

## Freeform Search

Database:

US Pre-Grant Publication Full-Text Database  
 US Patents Full-Text Database  
 US OCR Full-Text Database  
 EPO Abstracts Database  
 JPO Abstracts Database  
 Derwent World Patents Index  
 IBM Technical Disclosure Bulletins

Term:

L4 and (server\$ same (client\$ or user\$))

Display:

50

Documents in Display Format:

-

Starting with Number

1

 Generate: ☐ Hit List ☒ Hit Count ☐ Side by Side ☐ Image

Search

Clear

Interrupt

### Search History

 DATE: Wednesday, April 28, 2004 [Printable Copy](#) [Create Case](#)

Set Name   Query  
 side by side

Hit Count   Set Name  
 result set

DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR

<u>L5</u>	L4 and (server\$ same (client\$ or user\$))	28	<u>L5</u>
<u>L4</u>	L3 and (access near level)	29	<u>L4</u>
<u>L3</u>	L2 and (access near right\$)	70	<u>L3</u>
<u>L2</u>	L1 and (access\$ near information)	390	<u>L2</u>
<u>L1</u>	event near management	1739	<u>L1</u>

END OF SEARCH HISTORY

[First Hit](#)   [Fwd Refs](#)

Generate Collection

Print

L5: Entry 12 of 28

File: USPT

Mar 2, 2004

DOCUMENT-IDENTIFIER: US 6701345 B1

TITLE: Providing a notification when a plurality of users are altering similar data in a health care solution environment

Abstract Text (1):

A notification when multiple users attempt to alter the same data may first begin when connections to a plurality of user stations are monitored. An instruction for initiating a load process is received from a user station. Data is downloaded from the one of the user stations to the server. It is determined whether another load process is being concurrently executed by another user station. If it is determined that a load process is being concurrently executed, a notification is sent to the user station. A notification is also sent to the user station that initiated the concurrently executing load process. At least one of the load processes is suspended upon detecting the concurrently executed load process. At least one of the load processes may be allowed to continue upon receiving a command to continue from the user station associated with the suspended load process.

Brief Summary Text (19):

A notification when multiple users attempt to alter the same data may first begin when connections to a plurality of user stations are monitored. An instruction for initiating a load process is received from one of the user stations. Data is downloaded from the one of the user stations to the server. It is determined whether another load process is being concurrently executed by another user station. If it is determined that a load process is being concurrently executed, a notification is sent to the one of the user stations. A notification is also sent to the user station that initiated the concurrently executing load process. Both users are notified to allow them to coordinate their updates so that all alterations to the data are entered. At least one of the load processes is suspended upon detecting the second concurrently executed load process to allow the users time to react to the notification upon it being determined that another load process is being concurrently executed. One of the load processes, all but the first load process, all of the load processes, or any other combination can be suspended. At least one of the load processes may be allowed to continue upon receiving a command to continue from the user station associated with the suspended at least one of the load processes.

Drawing Description Text (5):

FIG. 3 is a flowchart depicting a process for providing a multi-tier client/server architecture for storing files and/or records;

Drawing Description Text (7):

FIG. 5 depicts a process for providing status messaging during data loading in a multi-tier client/server architecture;

Drawing Description Text (9):

FIG. 7 is a flowchart illustrating a process for loading data in a multi-tier client/server architecture;

Detailed Description Text (40):

To date, Web development tools have been limited in their ability to create dynamic

Web applications which span from client to server and interoperate with existing computing resources. Until recently, HTML has been the dominant technology used in development of Web-based solutions. However, HTML has proven to be inadequate in the following areas:

Detailed Description Text (51):

Sun's.RTM. Java.RTM. language has emerged as an industry-recognized language for "programming the Internet." Sun defines Java as: "a simple, object-oriented, distributed, interpreted, robust, secure, architecture-neutral, portable, high-performance, multithreaded, dynamic, buzzword-compliant, general-purpose programming language. Java supports programming for the Internet in the form of platform-independent Java applets." Java applets are small, specialized applications that comply with Sun's Java Application Programming Interface (API) allowing developers to add "interactive content" to Web documents (e.g., simple animations, page adornments, basic games, etc.). Applets execute within a Java-compatible browser (e.g., Netscape Navigator) by copying code from the server to client. From a language standpoint, Java's core feature set is based on C++. Sun's Java literature states that Java is basically, "C++ with extensions from Objective C for more dynamic method resolution."

Detailed Description Text (71):

FIG. 2 illustrates a data load process 200 in which a single user runs the process on an individual client desktop 202 (user station). An illustrative data load process may be embodied in a three tier client/server architecture including a Graphical User Interface (GUI) built in Microsoft Access, a server application built in C, Pro\*C, Perl 5 and Unix korn shell scripts, Oracle SQL\*Loader scripts, and a series of Oracle PL/SQL stored procedures.

Detailed Description Text (72):

In the data load process, a user logs onto the system. See arrow ref 208. As shown at arrow ref 210, the user selects specific keywords within a tier 204 to load into the database 206. The user executes a load process at arrow ref 212 and files to be loaded are transferred to the server at arrow ref 214. A load process control module is executed and the corresponding DMT(s) for the selected keyword(s) are sent to the server application. See arrow ref 216. A check for concurrently executing load processes is performed in operation 218. The success of the file transfer is performed in operation 220. In operations 222 and 224, the files are reformatted and loaded into tables by the server application loads data into worktables. The server application initiates stored PL/SQL procedures to perform validation. See operation 226. Data is validated according to database and/or client-specific business rules. If no validation errors are found, data is loaded into the Diamond database. See operation 228. If errors are found, a file containing all the good records, and a file containing all the bad records are sent back to the client desktop. See arrow ref 230. A report is produced listing all the erred records and the corresponding row numbers and error messages. Also, a verification report is produced that provides control totals for data loaded into database, or written to good/bad files. The reports can then be reviewed by the user. See arrow 232.

Detailed Description Text (74):

FIG. 3 is a flowchart depicting a process 300 for providing a multi-tier client/server architecture for storing files and/or records such as medical records. In operation 302, a connection is maintained between multiple user stations and a server that has a database. In this and the other embodiments set forth herein, the connection may be maintained utilizing a local area network or a wide area network. Alternatively, a dialup connection could be created periodically or upon user request. A plurality of records/files and a command to load the records into the database are received from one of the user stations in operation 304. The command may be ordered by the user, or may be executed automatically. If the command is executed automatically, it may be performed at predetermined

intervals. In operation 306, a data management template corresponding to the files/records is selected. The data management template may include a listing of all records/files that should be loaded. Alternatively, the data management template may specify particular content of the files/records that must be matched for verification. As another option, the data management template may specify specific particular sizes of the files/records. In operation 308, it is validated that all of the records/files to be loaded match the data management template. In operation 310, the records/files are sent to a database for loading in the database upon validation that the records match the data management template.

Detailed Description Text (81):

FIG. 5 depicts a process 500 for providing status messaging during data loading in a multi-tier client/server architecture. In operation 502, data is downloaded from a user station. A status of the download of the data is transmitted to the user station in operation 504. Preferably, the status is displayed as it is received. In operation 506, the data is divided into divisible portions. Each of the divisible portions of the data is checked in operation 508 to validate that the data meets predetermined criteria, such as that it includes certain content. In operation 510, a message is sent to the user station indicating whether the divisible portions of the data meet the predetermined criteria. The data is loaded in a database in operation 512. The data may include medical records.

Detailed Description Text (87):

FIG. 7 is a flowchart illustrating a process 700 for loading data in a multi-tier client/server architecture. In operation 702, a plurality of user-selected keywords are received. Data is organized around the keywords. The data can include medical-related data such as medical records. A data management template which corresponds to the keywords is selected in operation 704. A validation is performed in operation 706 to determine whether all of the data to be loaded matches the data management template. The data is sent to a database in operation 708 to be loaded in the database upon validation that the data matches the data management template.

Detailed Description Text (136):

FIG. 10 is an illustration showing a security organization according to one embodiment of the present invention. A Security Management Team may have a security management 1000, under which are an administration team 1002, a projects & planning team 1004, and a business process security team 1006. The size of the Security Management team, and the way in which it is integrated into the development organization depends on the degree to which security is a factor for each specific environment. For example, the security risks associated with an Internet-based online banking system are far greater than those of a fully isolated client/server system, and therefore warrant a larger team with broader responsibilities and greater influence.

Detailed Description Text (328):

It is important to set up and communicate a detailed folder structure with specified access rights from the beginning. Contents of folders must be checked regularly to ensure that folders contain what they are supposed to.

Detailed Description Text (333):

Another important distinction is the one between work in progress and completed documents that have been approved. This distinction can be supported by a folder structure with carefully chosen access rights.

Detailed Description Text (621):

Although direct sabotage is rare, inexperienced developers, perhaps new to the project, can wreak havoc to the system under development by inadvertently deleting or modifying system components. Focus must be on defining access rights so that developers have the right level of access (read/write) to all the information that

is useful and relevant to their work.

Detailed Description Text (804):

Assembly Test--The assembly test tests the interaction of related components to ensure that the components, when integrated, function properly. Assembly test ensures that data is passed correctly between screens in a conversation or batch process and that messages are passed correctly between a client and a server. The specification tested is the technical design. The application flow diagram within the technical design depicts the assemblies, either on-line conversations or batch assemblies, that will be assembly tested. Testing is therefore organized by assembly rather than by business function.

Detailed Description Text (818):

The Operational Readiness Test--The objective of the operational readiness test is to ensure that the application can be correctly deployed. The operational readiness test is also commonly known as the readiness test, roll-out test, release test, or the conversion test. The operational readiness test becomes especially key in client/server environments. It has four parts: Roll out test--ensures that the roll out procedures and programs can install the application in the production environment. Operations test--ensures that all operational procedures are in place and acceptable, and that the production system can be operated by the personnel responsible for supporting production. Service level test--ensures that once the application is rolled out, it provides the level of service to the users as specified in the Service Level Agreement (SLA). Roll out verification--ensures that the application has been correctly rolled out at each site. This test, developed by the work cell or team performing operational readiness test, should be executed during each site installation by the work cell or team in charge of the actual roll out of the application.

Detailed Description Text (942):

When processes become complex and require the participation of multiple groups, simple integration techniques are not adequate for managing the process flow. Workflow Management tools address this problem by providing the ability to define, manage, and execute automated business processes through an electronic representation of the process, both in terms of what has to be done, and by whom. For any process where multiple groups are involved, well-defined procedures must be in place to ensure that work flows from one task to another. Each participant must have access to the information required to perform the task, including the information from previous steps in the flow. This can be handled manually or supported by tools. If handled manually, it requires dedication, attention to detail, and significant training.

Detailed Description Text (987):

Role-based access control establishes access rights and profiles based on job functions within the environment. If different access rights are required for security administrators vs. code developers vs. code reviewers vs. testers, then the correct access can be established based on these functions.

Detailed Description Text (1026):

Repository access control is important where developers in the development environment need to be assigned different rights to the repository. Typically, the developers will be placed in groups with diminishing access rights such as repository administrator, technical support, designer, or programmer. These access rights may relate to read/write/modify/delete authority. This method of access control is far more flexible than simple object locking.

Detailed Description Text (1064):

Repository access can sometimes be controlled using an access control function, which comes with the repository. A common technique is to group users and assign different access rights to the different groups. Each of these groups is also

assigned specific read/write/delete/modify authority. For example, the following groups may be defined as having increasing rights:

Detailed Description Text (1083):

Flexible access rights based on user profiles, which differentiate (at least) between read and write access

Detailed Description Text (1176):

g) Does the tool provide ease of access to information?

Detailed Description Text (1317):

Security tools are required in the development environment to ensure against unauthorized access by individuals and system processes, to limit damages caused by such unauthorized access, and to audit access the environment services. At the security management level, it may be valuable to have tools which help manage security profiles, security groups, and access rights.

Detailed Description Text (1320):

Role-based access control establishes access rights and profiles based on job functions within the environment. If different access rights are required for security administrators vs. code developers vs. code reviewers vs. testers, then the correct access can be established based on these functions.

Detailed Description Text (1335):

Performance modeling tools in this category support the analysis of the development environment's performance, as opposed to that of the client/server application being developed. A simple spreadsheet may be suitable in some well-known and understood environments, but dedicated performance modeling tools should be considered on any project with high transaction volumes or complex environments involving multiple platforms.

Detailed Description Text (1366):

Test Data Management--Test results, expected results, and data comparison results can be linked to a defect to provide centralized access to the information. Integration also aids in keeping track of the cycle where the problem occurred, the test condition, and therefore the business function affected by the problem.

Detailed Description Text (1385):

Design tools are used to specify "how" a system will implement these system requirements. They are typically diagramming tools, which graphically depict how the system will be built in terms of its key components. This differs between classical client/server systems and component-based systems:

Detailed Description Text (1386):

The standard client/server model comprises application logic, presentation, and communication components, which together support the business processes. For a client/server system, each of these components must be individually defined.

Detailed Description Text (1441):

The information management component may provide the security needed in a multi-designer environment. If this is not the case then a multi-designer data modeling tool should be used. The tool may provide a central dictionary which allows design data to be shared between several designers and includes security checks to monitor any conflicts in overlapping access rights between designers.

Detailed Description Text (1515):

c) Are there hundreds of users? Are there tens of servers?

Detailed Description Text (1634):

Application Logic Design tools are used to graphically depict an application. These

tools include application structure, module descriptions, and distribution of functions across client/server nodes.

Detailed Description Text (1635):

A variety of tools and techniques can be used for Application Logic Design. Examples are structure charts, procedure diagrams (module action diagrams), and graphics packages to illustrate distribution of functions across client and server.

Detailed Description Text (1755):

Communication design tools are essential in developing systems where critical business operations have to have maximum availability and minimum down time. One of the primary contributing factors to high performance in client/server environments is a good network design. A good network design can only be achieved through a good communication design.

Detailed Description Text (1779):

Reverse engineering tools are used to capture specific, relevant functional and design information from a legacy system for use in a new, client/server system or to restructure the existing system for improved performance and maintenance.

Detailed Description Text (1813):

Construction tools are used to program or build the application: client and server source code, windows, reports, and database. Along with the onset of Visual Programming, the more traditional form of construction tools have been superceded by Integrated Development Environments (IDEs) which take all the basic components required for construction, and integrate them into a single system. Although IDEs are now the preferred tools for most construction, the components that make up these tools remain the same--Source Code Editor, Compiler/Linker/Interpreter, Generation Tools and Debugging Tools.

Detailed Description Text (1814):

Visual Programming tools, initially associated with the rapid development of the client-side of client/server applications, have now matured and expanded their domain to cover entire client/server development (e.g. Visual C++) and Netcentric development (e.g. visual Java IDEs).

Detailed Description Text (1932):

Testing applications (client/server or Netcentric) remains a complex task because of the large number of integrated components involved (for example, multiplatform clients, multiplatform servers, multitiered applications, communications, distributed processing, and data), which, in turn, results in a large number and variety of Testing tools.

Detailed Description Text (2167):

Event Management (2018)

Detailed Description Text (2168):

An event is an electronic message generated by any component (e.g., application software, system software, hardware, etc.) in the system. Event Management receives, logs, classifies and presents event messages on a console(s) based on pre-established filters or thresholds.

Detailed Description Text (2211):

Automatic logging of problems will require interfaces to be built with the Event Management system, and perhaps the execution architecture for application errors.

Detailed Description Text (2225):

The way in which a disaster is defined will be dependent upon which resources are critical to the business. For example, a data center failure may be critical for

one client whereas a server failure for another is more critical.

Detailed Description Text (2391):

Backups are typically embedded into production scheduling with restores on an ad hoc basis. Backup/Restore needs to ensure that a file can be only backed up/restored by users with the right access level. Furthermore, file transfer utilities need to be used when the information to archived is sent through the network as well as security for file control access and global authorization should be available and done in concert with the security management facility.

Detailed Description Text (2643):

Capacity Planning & Modeling must coordinate the requirements across the system (e.g., networks, servers, workstations, CPU, etc.) Capacity is driven by the need to meet SLAs with the user communities and as part of the planning and modeling process, future threats to capacity should be identified.

Detailed Description Text (2732):

Managing hardware is all hardware directly used to manage the environment. This includes all staging components. These components are devoted to systems management functions. Examples of managing hardware include management servers, management controllers, management consoles, probes, and sniffers. One significant component in the hardware monitoring arena is Firewall access control policy management. Firewalls are regularly used for network based security management. It is typically a system or group of systems that enforce access control between two or more networks and/or perform network data packet filtering. Usually packet filtering router hardware and application gateways are used to block unauthorized IP packets and enforce proxy defined user commands.

Detailed Description Text (2753):

Event Management

Detailed Description Text (2754):

An event is an electronic message generated by any component (e.g., application software, system software, hardware, etc.) in the system. Event Management receives, logs, classifies and presents event messages on a console(s) based on pre-established filters or thresholds.

Detailed Description Text (2757):

The scope of events to be monitored will have a major impact on the approach taken for Event management and the tools selected.

Detailed Description Text (2765):

The number of events generated in the system will increase due to the complexity of the system. Devices will generate events as well as applications, the technical infrastructure, etc. Common event handling mechanisms will be required to provide management information in a simple, consistent format and to forward important events on for management purposes. In addition, filtering capabilities may also be needed at remote locations to prevent the streaming of events to central/master management consoles.

Detailed Description Text (2774):

The physical environment includes all the support indirectly involved in maintaining and managing the distributed environment. Initially it was thought client/server technology would make data centers obsolete. However, with the migration of mission critical processes to client/server environments, many servers are being maintained in data centers in an effort to increase reliability. As a result, the importance of managing the physical environment has increased. Partially because it was initially believed not to be very important and because it does not relate directly to the information systems, the physical environment of the operational architecture is often overlooked. These systems include UPS, raised



floor, power, site survey and preparation, wiring/cabling, climate control, etc.

CLAIMS:

1. A method for providing a notification when multiple users attempt to alter the same data, comprising the steps of: (a) monitoring connections to a plurality of user stations; (b) receiving an instruction from a first user stations for initiating a first load process for loading data from said first user station to a server; (c) downloading the data to be loaded from the first user station to the server in a first load process; (d) after said downloading, determining whether a second load process is being concurrently executed by a second user station; (e) sending a notification to the first user stations if it is determined that a second load process is being concurrently executed; (f) sending a notification to the second user station; (g) suspending at least one of the first or second load processes upon it being determined that the second load process is being concurrently executed; and (h) allowing the suspended load processes to continue upon receiving a command to continue from user stations initiating the suspended load processes.

7. A computer program embodied on a computer readable medium for providing a notification when multiple users attempt to alter the same data, comprising: (a) a code segment that monitors connections to a plurality of user stations; (b) a code segment that receives an instruction from a first user station for initiating a first load process for loading data from said first user station to a server; (c) a code segment that downloads the data to be loaded from the first user station to the server in a first load process; (d) a code segment that determines after said downloading, whether a second load process is being concurrently executed by a second user station; (e) a code segment that sends a notification to the first user station (f) a code segment that sends a notification to the second user station; (g) a code segment that suspends at least one of the first or second load processes upon it being determined that the second load process is being concurrently executed; and (h) a code segment that allows the suspended load processes to continue upon receiving a command to continue from user stations initiating the suspended load processes.

13. A system for providing a notification when multiple users attempt to alter the same data, comprising: (a) logic that monitors connections to a plurality of user stations; (b) logic that receives an instruction from a first user stations for initiating a first load process for loading data from said first user station to a server; (c) logic that downloads the data to be loaded from the first user stations to the server in a first load process; (d) logic that determines after said downloading, whether a second load process is being concurrently executed by a second user station; (e) logic that sends a notification to the first user stations if it is determined that a secondload process is being concurrently executed; (f) logic that sends a notification to the second user station; (g) logic that suspends at least one of the first or second load processes upon it being determined that the second load process is being concurrently executed; and (h) logic that allows the suspended load processes to continue upon receiving a command to continue from user stations initiating the suspended load processes.

## Hit List

[Clear](#) [Generate Collection](#) [Print](#) [Fwd Refs](#) [Bkwd Refs](#)  
[Generate OACS](#)

Search Results - Record(s) 1 through 10 of 10 returned.

☐ 1. Document ID: US 20040006652 A1

Using default format because multiple data bases are involved.

L3: Entry 1 of 10

File: PGPB

Jan 8, 2004

PGPUB-DOCUMENT-NUMBER: 20040006652

PGPUB-FILING-TYPE: new

DOCUMENT-IDENTIFIER: US 20040006652 A1

TITLE: System event filtering and notification for OPC clients

PUBLICATION-DATE: January 8, 2004

INVENTOR-INFORMATION:

NAME	CITY	STATE	COUNTRY	RULE-47
Prall, John M.	Cave Creek	AZ	US	
Lin, Haur J.	Phoenix	AZ	US	
Urso, Jason T.	Cave Creek	AZ	US	

US-CL-CURRENT: 719/318

Full	Title	Citation	Front	Review	Classification	Date	Reference	Sequences	Attachments	Claims	FIGS	Draw. D.
------	-------	----------	-------	--------	----------------	------	-----------	-----------	-------------	--------	------	----------

☐ 2. Document ID: US 20020016729 A1

L3: Entry 2 of 10

File: PGPB

Feb 7, 2002

PGPUB-DOCUMENT-NUMBER: 20020016729

PGPUB-FILING-TYPE: new

DOCUMENT-IDENTIFIER: US 20020016729 A1

TITLE: System and method for scheduling events and associated products and services

PUBLICATION-DATE: February 7, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	COUNTRY	RULE-47
Breitenbach, Matthew	Philadelphia	PA	US	
Low, Marty	Collegeville	PA	US	
Matranga, John J.	Berwyn	PA	US	

US-CL-CURRENT: 705/9

Full	Title	Citation	Front	Review	Classification	Date	Reference	Sequences	Attachments	Claims	KWIC	Draw Ds
------	-------	----------	-------	--------	----------------	------	-----------	-----------	-------------	--------	------	---------

☐ 3. Document ID: US 6718535 B1

L3: Entry 3 of 10

File: USPT

Apr 6, 2004

US-PAT-NO: 6718535

DOCUMENT-IDENTIFIER: US 6718535 B1

TITLE: System, method and article of manufacture for an activity framework design in an e-commerce based environment

DATE-ISSUED: April 6, 2004

## INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Underwood; Roy Aaron	Long Grove	IL		

## ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Accenture LLP	Palo Alto	CA			02

APPL-NO: 09/ 364164 [PALM]

DATE FILED: July 30, 1999

INT-CL: [07] G06 F 9/44

US-CL-ISSUED: 717/101; 717/120

US-CL-CURRENT: 717/101; 717/120

FIELD-OF-SEARCH: 717/116, 717/100, 717/101, 717/102, 717/108, 717/165, 717/120, 717/223

PRIOR-ART-DISCLOSED:

## U.S. PATENT DOCUMENTS

PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
<u>4370707</u>	January 1983	Phillips et al.	
<u>5333302</u>	July 1994	Hensley et al.	
<u>5371852</u>	December 1994	Attanasio et al.	
<u>5423032</u>	June 1995	Byrd et al.	
<u>5437027</u>	July 1995	Bannon et al.	
<u>5440719</u>	August 1995	Hanes et al.	
<u>5530829</u>	June 1996	Beardsley et al.	
<u>5623601</u>	April 1997	Vu	
<u>5630069</u>	May 1997	Flores et al.	
<u>5630131</u>	May 1997	Palevich et al.	

<u>5659735</u>	August 1997	Parrish et al.	
<u>5666502</u>	September 1997	Capps	
<u>5673386</u>	September 1997	Batra	
<u>5673387</u>	September 1997	Chen et al.	
<u>5677997</u>	October 1997	Talatik	706/45
<u>5692132</u>	November 1997	Hogan	
<u>5694540</u>	December 1997	Humelsine et al.	
<u>5710884</u>	January 1998	Dedrick	
<u>5724521</u>	March 1998	Dedrick	
<u>5740427</u>	April 1998	Stoller	
<u>5754173</u>	May 1998	Hiura et al.	
<u>5754938</u>	May 1998	Herz et al.	
<u>5754939</u>	May 1998	Herz et al.	
<u>5758062</u>	May 1998	McMahon et al.	
<u>5758074</u>	May 1998	Marlin et al.	
<u>5764897</u>	June 1998	Khalidi	709/201
<u>5778169</u>	July 1998	Reinhardt	
<u>5784553</u>	July 1998	Kolawa et al.	
<u>5812669</u>	September 1998	Jenkins et al.	
<u>5815657</u>	September 1998	Bipinkumar et al.	
<u>5819265</u>	October 1998	Ravin et al.	
<u>5819281</u>	October 1998	Cummins	707/103
<u>5819295</u>	October 1998	Nakagawa et al.	
<u>5835087</u>	November 1998	Herz et al.	
<u>5835911</u>	November 1998	Nakagawa et al.	
<u>5844508</u>	December 1998	Murashita et al.	
<u>5872973</u>	February 1999	Mitchell et al.	709/332
<u>5878432</u>	March 1999	Misheski	
<u>5889520</u>	March 1999	Glaser	
<u>5890161</u>	March 1999	Helland et al.	
<u>5890175</u>	March 1999	Wong et al.	
<u>5892898</u>	April 1999	Fujii et al.	
<u>5930512</u>	July 1999	Boden et al.	
<u>5937165</u>	August 1999	Schwaller et al.	
<u>5949419</u>	September 1999	Domine et al.	
<u>5956732</u>	September 1999	Tsuchida	
<u>5956736</u>	September 1999	Hanson et al.	
<u>5960200</u>	September 1999	Eager et al.	717/147
<u>5995114</u>	November 1999	Wegman et al.	
<u>6014669</u>	January 2000	Slaughter et al.	
<u>6016495</u>	January 2000	Mckeehan et al.	707/103
<u>6029178</u>	February 2000	Martin et al.	
<u>6029195</u>	February 2000	Herz et al.	
<u>6035323</u>	March 2000	Narayan et al.	
<u>6044368</u>	March 2000	Powers	
<u>6055538</u>	April 2000	Kessenich et al.	
<u>6058260</u>	May 2000	Brokel et al.	

<u>6058379</u>	May 2000	Odom et al.	
<u>6061643</u>	May 2000	Walker et al.	
<u>6101503</u>	August 2000	Cooper et al.	
<u>6108670</u>	August 2000	Weida et al.	
<u>6112228</u>	August 2000	Earl et al.	
<u>6112240</u>	August 2000	Pogue et al.	
<u>6115544</u>	September 2000	Mueller	
<u>6137869</u>	October 2000	Voit et al.	
<u>6141010</u>	October 2000	Hoyle	
<u>6141647</u>	October 2000	Meijer et al.	
<u>6151600</u>	November 2000	Derick	
<u>6151610</u>	November 2000	Senn et al.	
<u>6167564</u>	December 2000	Fontana et al.	717/104
<u>6182226</u>	January 2001	Reid et al.	
<u>6185625</u>	February 2001	Tso et al.	
<u>6195794</u>	February 2001	Buxton	
<u>6199068</u>	March 2001	Carpenter	
<u>6199079</u>	March 2001	Gupta et al.	
<u>6202051</u>	March 2001	Woolston	
<u>6208345</u>	March 2001	Sheard et al.	345/356
<u>6209000</u>	March 2001	Klien et al.	
<u>6209033</u>	March 2001	Datta et al.	
<u>6222535</u>	April 2001	Hurd, II	
<u>6223221</u>	April 2001	Kunz	
<u>6230160</u>	May 2001	Chan et al.	
<u>6230194</u>	May 2001	Frailong et al.	
<u>6230309</u>	May 2001	Turner et al.	
<u>6233584</u>	May 2001	Purcell	
<u>6237114</u>	May 2001	Wookey et al.	
<u>6246410</u>	June 2001	Bergeron et al.	
<u>6249905</u>	June 2001	Yoshida et al.	
<u>6256659</u>	July 2001	McLain et al.	
<u>6256678</u>	July 2001	Traughber et al.	
<u>6260068</u>	July 2001	Zalewski et al.	
<u>6266666</u>	July 2001	Ireland et al.	
<u>6272673</u>	August 2001	Dale et al.	
<u>6272678</u>	August 2001	Imachi et al.	
<u>6282605</u>	August 2001	Moore	
<u>6286028</u>	September 2001	Cohen et al.	
<u>6301601</u>	October 2001	Heller et al.	
<u>6304893</u>	October 2001	Gish	
<u>6308188</u>	October 2001	Bernardo et al.	
<u>6313835</u>	November 2001	Gever et al.	
<u>6314434</u>	November 2001	Shigemi et al.	
<u>6327677</u>	December 2001	Garg et al.	
<u>6336118</u>	January 2002	Hammond et al.	
<u>6401085</u>	June 2002	Gershman et al.	

<u>6430556</u>	August 2002	Goldberg et al.	707/4
<u>6438514</u>	August 2002	Hill et al.	703/17
<u>6442620</u>	August 2002	Thatte et al.	
<u>6473794</u>	October 2002	Guheen et al.	709/223

## FOREIGN PATENT DOCUMENTS

FOREIGN-PAT-NO	PUBN-DATE	COUNTRY	US-CL
0398646	November 1990	EP	
0587394	March 1994	EP	
0643359	March 1995	EP	
WO 98/35297	August 1998	WO	
WO 98/35469	August 1998	WO	
WO 99/01826	January 1999	WO	
WO 01/09752	February 2001	WO	
WO 01/10082	February 2001	WO	

## OTHER PUBLICATIONS

Akerley, J., Li N., Parlavecchia A., Programming with VisualAge for Java Version 2, IBM Redbooks, Nov. 1998, pp. 53, 57, 112, 122 295-326.

Anonymous. Optimal Networks Partners With Shomiti Systems to Deliver Extended Application Analysis, press release dated Jul. 14, 1997, three-page text download from.

Behroozi-Toosi et al., "Modeling of Computer Networks and Systems Using SES/Workbench," Proceedings of the Midwest Symposium on Circuits and Systems, May 1991, vol. SYMP. 34, pp. 992-996.

Beta, M., Active Data Objects & ASP, Dr. Dobb's Journal, U.S. M&T Publ., May 1998, vol. 23, No. 5, pp. 88-91, 111-112.

Conradi et al., "Version Models for Software Configuration Management," ACM, pp. 232-282, Jun. 1998.

Dart, S., "Concepts in Configuration Management Systems," ACM, pp. 1-18, 1991.

Dror, A., R. Lafore, "OS/2 Presentation Manager Programming Primer," 1990. McGraw Hill, Berkeley California. Chapter 7, pp. 165-192.

Katzela, Modeling and Simulating Communication Networks: A Hands-O Approach Using OPNET, Mar. 1998, pp. 63-90.

Knutson, "Building the Model Network," Byte, Oct. 1996, vol. 21, No. 10, pp. 101-102, 104.

Law et al., "Stimulation Software for Communications Networks: The State of the Art," IEEE Communications Magazine, Mar. 1994, vol. 32, No. 3, pp. 44-50.

Limprecht, R., "Microsoft Transaction Server," IEEE, Feb. 25, 1997.

Lin et al., "Configuration Management with Logical Structures," ACM, pp. 298-307, 1996.

MIL 3 Inc., "OPNET Tool Operations Manual," 1998, pp. ii-viii, 21Nt-48Nt, 127Pb-143Pb.

Mishra et al., "Effect of Connection Rerouting on Application Performance in Mobile Networks," IEEE Transactions on Computers, vol. 47, No. 4, pp. 371-390. Apr. 1998.

Muller, "Design and Conquer," Byte, Oct. 1996, vol. 21, No. 10, pp. 93-94, 96, 98.

Van Norman, "WAN Design Tools: The New Generation Today's Wide-area Network Design Tools, with Enhanced Tariff Information, Analysis Capabilities and Functions, are Better Than Ever," Data Communications, Oct. 1990, vol. 19, No. 13, pp. 105-106, 108-110, 112.

O'Conner, J., "Internationalization: Localization with ResourceBundles," Java Developer Connection, Online!, Oct. 1998.

Rosove, "Training Requirements for Computer Programmers in Relation to System Development Phases," ACM, Proceedings of the Seventh Annual SIGCPR Conference, pp. 65-76, 1969.

Spitzer, "Tiers Without Tears," DBMS Online, Jan. 12, 1998, pp. 1-5.

Sun Microsystems: "I18N Resource Manager" Sun Documentation, Online!, Apr. 17, 1998, pp. 1-17.

Template Software Inc., Web Component Foundation Template, Using the Web Component (WEB), 1997, Version 8, Chapters 1-3.

Template Software Inc., Workflow Template Process Template Developing a WFT Workflow System (WFT), 1997, Version 8, Chapters 1-10.

Template Software Inc., "SNAP Foundation Template, Using the SNAP Development Environment (SNAP DEV)." 1997, Version 8, Chapters 1-11.

Vinoski, "Corba: Integrating Diverse Applications Within Distributed Heterogeneous Environments," IEEE Communications Magazine, Feb. 1997, pp. 46-55.

ART-UNIT: 2122

PRIMARY-EXAMINER: Nguyen-Ba; Hoang-Vu Anthony

ATTY-AGENT-FIRM: Edwards; W. Glenn Oppenheimer Wolff & Donnelly LLP

ABSTRACT:

A system and method are provided for providing an activity framework. First, a plurality of sub-activities are created which each include sub-activity logic adapted to generate an output based on an input received from a user upon execution. Second, a plurality of activities are defined which each execute the sub-activities in a unique manner upon being selected for accomplishing a goal associated with the activity. Selection of one of the activities is allowed by receiving user indicia. An interface is depicted for allowing receipt of the input and display of the output during execution of the sub-activities associated with the selected activity.

24 Claims, 179 Drawing figures

Full	Title	Citation	Front	Review	Classification	Date	Reference	Abstract	Claims	KWIC	Draw. De
------	-------	----------	-------	--------	----------------	------	-----------	----------	--------	------	----------

☐ 4. Document ID: US 6704873 B1

L3: Entry 4 of 10

File: USPT

Mar 9, 2004

US-PAT-NO: 6704873

DOCUMENT-IDENTIFIER: US 6704873 B1

TITLE: Secure gateway interconnection in an e-commerce based environment

DATE-ISSUED: March 9, 2004

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Underwood; Roy Aaron	Long Grove	IL		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Accenture LLP	Palo Alto	CA			02

APPL-NO: 09/ 364490 [PALM]

DATE FILED: July 30, 1999

INT-CL: [07] G06 F 11/30

US-CL-ISSUED: 713/201; 709/223, 709/249

US-CL-CURRENT: 713/201; 709/223, 709/249

FIELD-OF-SEARCH: 709/249, 709/223, 713/200-202

PRIOR-ART-DISCLOSED:

## U.S. PATENT DOCUMENTS

PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
<u>4370707</u>	January 1983	Phillips et al.	
<u>5333302</u>	July 1994	Hensley et al.	
<u>5371852</u>	December 1994	Attanasio et al.	
<u>5423032</u>	June 1995	Byrd et al.	
<u>5437027</u>	July 1995	Bannon et al.	
<u>5440719</u>	August 1995	Hanes et al.	
<u>5530829</u>	June 1996	Beardsley et al.	
<u>5623601</u>	April 1997	Vu	713/201
<u>5630069</u>	May 1997	Flores et al.	
<u>5630131</u>	May 1997	Palevich et al.	
<u>5659735</u>	August 1997	Parrish et al.	
<u>5666502</u>	September 1997	Capps	
<u>5673386</u>	September 1997	Batra	
<u>5673387</u>	September 1997	Chen et al.	
<u>5677997</u>	October 1997	Talatik	
<u>5692132</u>	November 1997	Hogan	
<u>5694540</u>	December 1997	Humelsine et al.	
<u>5710884</u>	January 1998	Dedrick	
<u>5724521</u>	March 1998	Dedrick	
<u>5740427</u>	April 1998	Stoller	
<u>5754173</u>	May 1998	Hiura et al.	
<u>5754938</u>	May 1998	Herz et al.	
<u>5754939</u>	May 1998	Herz et al.	
<u>5758062</u>	May 1998	McMahon et al.	
<u>5758074</u>	May 1998	Marlin et al.	
<u>5764897</u>	June 1998	Khalidi	
<u>5778169</u>	July 1998	Reinhardt	
<u>5784553</u>	July 1998	Kolawa et al.	
<u>5812669</u>	September 1998	Jenkins et al.	
<u>5815657</u>	September 1998	Parmar et al.	
<u>5819265</u>	October 1998	Ravin et al.	
<u>5819281</u>	October 1998	Cummins	



<u>5819295</u>	October 1998	Nakagawa et al.	
<u>5835087</u>	November 1998	Herz et al.	
<u>5835911</u>	November 1998	Nakagawa et al.	
<u>5844508</u>	December 1998	Murashita et al.	
<u>5872973</u>	February 1999	Mitchell et al.	
<u>5878432</u>	March 1999	Misheski et al.	
<u>5889520</u>	March 1999	Glaser	
<u>5890161</u>	March 1999	Helland et al.	
<u>5890175</u>	March 1999	Wong et al.	
<u>5892898</u>	April 1999	Fujii et al.	
<u>5930512</u>	July 1999	Boden et al.	
<u>5937165</u>	August 1999	Schwaller et al.	
<u>5949419</u>	September 1999	Domine et al.	
<u>5956732</u>	September 1999	Tsuchida	
<u>5956736</u>	September 1999	Hanson et al.	
<u>5960200</u>	September 1999	Eager et al.	
<u>5995114</u>	November 1999	Wegman et al.	
<u>6014669</u>	January 2000	Slaughter et al.	
<u>6016495</u>	January 2000	McKeehan et al.	
<u>6029178</u>	February 2000	Martin et al.	
<u>6029195</u>	February 2000	Herz	
<u>6035323</u>	March 2000	Narayen et al.	
<u>6044368</u>	March 2000	Powers	
<u>6055538</u>	April 2000	Kessenich et al.	
<u>6058260</u>	May 2000	Brokel et al.	
<u>6058379</u>	May 2000	Odom et al.	
<u>6061643</u>	May 2000	Walker et al.	
<u>6079020</u>	June 2000	Liu	713/201
<u>6101503</u>	August 2000	Cooper et al.	
<u>6108670</u>	August 2000	Weida et al.	
<u>6112228</u>	August 2000	Earl et al.	
<u>6112240</u>	August 2000	Pogue et al.	
<u>6115544</u>	September 2000	Mueller	
<u>6137869</u>	October 2000	Voit et al.	370/238
<u>6141010</u>	October 2000	Hoyle	
<u>6141647</u>	October 2000	Meijer et al.	
<u>6151600</u>	November 2000	Derick	
<u>6151610</u>	November 2000	Senn et al.	
<u>6167564</u>	December 2000	Fontana et al.	
<u>6182226</u>	January 2001	Reid et al.	709/225
<u>6185625</u>	February 2001	Tso et al.	
<u>6195794</u>	February 2001	Buxton	
<u>6199068</u>	March 2001	Carpenter	
<u>6199079</u>	March 2001	Gupta et al.	
<u>6202051</u>	March 2001	Woolston	
<u>6208345</u>	March 2001	Sheard et al.	
<u>6209000</u>	March 2001	Klien et al.	

<u>6209033</u>	March 2001	Datta et al.	
<u>6222535</u>	April 2001	Hurd, II	
<u>6223221</u>	April 2001	Kunz	
<u>6230160</u>	May 2001	Chan et al.	
<u>6230194</u>	May 2001	Frailong et al.	709/220
<u>6230309</u>	May 2001	Turner et al.	
<u>6233584</u>	May 2001	Purcell	
<u>6237114</u>	May 2001	Wookey et al.	
<u>6246410</u>	June 2001	Bergeron et al.	
<u>6249905</u>	June 2001	Yoshida et al.	
<u>6256659</u>	July 2001	McLain, Jr. et al.	
<u>6256678</u>	July 2001	Traughber et al.	
<u>6260068</u>	July 2001	Zalewski et al.	
<u>6266666</u>	July 2001	Ireland et al.	
<u>6272673</u>	August 2001	Dale et al.	
<u>6272678</u>	August 2001	Imachi et al.	
<u>6282605</u>	August 2001	Moore	
<u>6286028</u>	September 2001	Cohen et al.	
<u>6301601</u>	October 2001	Heller et al.	
<u>6304893</u>	October 2001	Gish	
<u>6308188</u>	October 2001	Bernardo et al.	
<u>6313835</u>	November 2001	Gever et al.	
<u>6314434</u>	November 2001	Shigemi et al.	
<u>6327677</u>	December 2001	Garg et al.	
<u>6336118</u>	January 2002	Hammond et al.	
<u>6401085</u>	June 2002	Gershman et al.	
<u>6430556</u>	August 2002	Goldberg et al.	
<u>6438514</u>	August 2002	Hill et al.	
<u>6442620</u>	August 2002	Thatte et al.	

## FOREIGN PATENT DOCUMENTS

FOREIGN-PAT-NO	PUBN-DATE	COUNTRY	US-CL
0398646	November 1990	EP	
0587394	March 1994	EP	
0643359	March 1995	EP	
WO 98/35297	August 1998	WO	
WO 98/35469	August 1998	WO	
WO 99/01826	January 1999	WO	
WO 01/09752	February 2001	WO	
WO 01/10082	February 2001	WO	

## OTHER PUBLICATIONS

"SunHIPPI 1.0 User's Installation Guide", Mar. 1999, pp. 69-70 (Glossary).  
[sunsolve.sun.com/data/805/808-7133/pdf/014.glossary.pdf](http://sunsolve.sun.com/data/805/808-7133/pdf/014.glossary.pdf).  
 "Internet Firewalls Frequently Asked Questions" by Marcus J. Ranum and Matt Curtin,  
 May 26, 1998. <http://195.195.128.170/staff/scotty/firewallfaq/fwfaq.html>.

Akerley, J., Li N., Parlavecchia A., Programming with VisualAge for Java Version 2, IBM Redbooks, Nov. 1998, pp. 53, 57, 112, 122 295-326.

Anonymous. Optimal Networks Partner With Shomiti Systems to Deliver Extended Application Analysis, press release dated Jul. 14, 1997, three-page text download from <http://www.shomiti.com/news/pr/1997/071497.html>.

Behroozi-Toosi et al., "Modeling of Computer Networks and Systems Using SES/Workbench," Proceedings of the Midwest Symposium on Circuits and Systems, May 1991, vol. SYMP. 34, pp. 992-996.

Beta, M., Active Data Objects & ASP, Dr. Dobb's Journal, U.S. M&T Publ., May 1998, vol. 23, No. 5, pp. 88-91, 111-112.

Conradi et al., "Version Models for Software Configuration Management," ACM, pp. 232-282, Jun. 1998.

Dart, S., "Concepts in Configuration Management Systems," ACM, pp. 1-18, 1991.

Dror, A., R. Lafore, "OS/2 Presentation Manager Programming Primer," 1990. McGraw Hill, Berkeley California. Chapter 7, pp. 165-192.

Katzela, Modeling and Simulating Communication Networks: A Hands-O Approach Using OPNET, Mar. 1998, pp. 63-90.

Knutson, "Building the Model Network," Byte, Oct. 1996, vol. 21, No. 10, pp. 101-102, 104.

Law et al., "Simulation Software for Communications Networks: The State of the Art," IEEE Communications Magazine, Mar. 1994, vol. 32, No. 3, pp. 44-50.

Limprecht, R., "Microsoft Transaction Server," IEEE, Feb. 25, 1997.

Lin et al., "Configuration Management with Logical Structures," ACM, pp. 298-307, 1996.

MIL 3 Inc., "OPNET Tool Operations Manual," 1998, pp. ii-viii, 21Nt-48Nt, 127Pb-143Pb.

Mishra et al., "Effect of Connection Rerouting on Application Performance in Mobile Networks," IEEE Transactions on Computers, vol. 47, No. 4, pp. 371-390, Apr. 1998.

Muller, "Design and Conquer," Byte, Oct. 1996, vol. 21, No. 10, pp. 93-94, 96, 98.

Norman, "WAN Design Tools: The New Generation Today's Wide-area Network Design Tools, with Enhanced Tariff Information, Analysis Capabilities, and Functions, are Better Than Ever," Data Communications, Oct. 1990, vol. 19, No. 13, pp. 105-106, 108-110, 112.

O'Conner, J., "Internationalization: Localization with ResourceBundles," Java Developer Connection, Online!, Oct. 1998  
(URL:<http://developer.java.sun.com/developer/technicalArticles/Int/ResourceBundles/index.html>, retrieved Apr. 4, 2001, pp. 1-5.

Rosove, "Training Requirements for Computer Programmers in Relation to System Development Phases," ACM, Proceedings of the Seventh Annual SIGCPR Conference, pp. 65-76, 1969.

Spitzer, "Tiers Without Tears," DBMS Online, Jan. 12, 1998, pp. 1-5.

Sun Microsystems: "I18N Resource Manager" Sun Documentation, Online!, Apr. 17, 1998, pp. 1-17 (URL:<http://falconet.inria.fr/{java/tools/irm/>, retrieved Apr. 4, 2001, pp. 4-17).

Template Software Inc., Web Component Foundation Template, Using the Web Component (WEB), 1997, Version 8, Chapters 1-3.

Template Software Inc., Workflow Template Process Template Developing a WFT Workflow System (WFT), 1997, Version 8, Chapters 1-10.

Template Software Inc., "SNAP Foundation Template, Using the SNAP Development Environment (SNAP DEV)." 1997, Version 8, Chapters 1-11.

Vinoski, "Corba: Integrating Diverse Applications Within Distributed Heterogeneous Environments," IEEE Communications Magazine, Feb. 1997, pp. 46-55.

ART-UNIT: 2143

PRIMARY-EXAMINER: Wiley; David

ASSISTANT-EXAMINER: Neurauter; George

ATTY-AGENT-FIRM: Edwards; W. Glenn Oppenheimer Wolff & Donnelly, LLP

## ABSTRACT:

A system and method of providing a global internetworking gateway architecture in an e-commerce environment are provided. A plurality of gateways each situated in a distinct geographic location are coupled to an internet. A wide area network, separate from the internet, is coupled to each of the gateways for providing communication between the wide area network and the internet. Coupled to the wide area network is a central database for providing a central storage for data used in e-commerce carried out over the internet. In one embodiment, at least one of the gateways includes at least one screening router coupled to the internet service provider, at least one firewall connected to the screening router, and a choker router coupled between the wide area network and the firewall.

16 Claims, 179 Drawing figures

Full	Title	Citation	Front	Review	Classification	Date	Reference			Claims	RWMC	Draw. De
------	-------	----------	-------	--------	----------------	------	-----------	--	--	--------	------	----------

☐ 5. Document ID: US 6701514 B1

L3: Entry 5 of 10

File: USPT

Mar 2, 2004

US-PAT-NO: 6701514

DOCUMENT-IDENTIFIER: US 6701514 B1

TITLE: System, method, and article of manufacture for test maintenance in an automated scripting framework

DATE-ISSUED: March 2, 2004

## INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Haswell; John Jeffrey	Herndon	VA		
Young; Robert J.	Charlestown	MA		
Schramm; Kevin	Rose Valley	PA		

## ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Accenture LLP	Palo Alto	CA			02

APPL-NO: 09/ 536264 [PALM]

DATE FILED: March 27, 2000

INT-CL: [07] G06 F 9/44, G06 F 7/00

US-CL-ISSUED: 717/115; 717/124, 707/102

US-CL-CURRENT: 717/115; 707/102, 717/124

FIELD-OF-SEARCH: 717/124, 717/126, 717/127, 717/128, 717/140, 717/149, 717/115, 707/102

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
<u>5119307</u>	June 1992	Blaha et al.	
<u>5627886</u>	May 1997	Bowman	
<u>5754760</u>	May 1998	Warfield	
<u>5781720</u>	July 1998	Parker et al.	
<u>5805891</u>	September 1998	Bizuneh et al.	717/121
<u>5870719</u>	February 1999	Maritzen et al.	
<u>5905856</u>	May 1999	Ottensooser	
<u>5954829</u>	September 1999	McLain, Jr. et al.	
<u>5960196</u>	September 1999	Carrier, III et al.	
<u>5983001</u>	November 1999	Boughner et al.	
<u>6002869</u>	December 1999	Hinckley	
<u>6002871</u>	December 1999	Duggan et al.	
<u>6006230</u>	December 1999	Ludwig et al.	
<u>6014760</u>	January 2000	Silva et al.	
<u>6031990</u>	February 2000	Sivakumar et al.	
<u>6058492</u>	May 2000	Sample et al.	
<u>6064381</u>	May 2000	Harel	
<u>6067639</u>	May 2000	Rodrigues et al.	714/38
<u>6069873</u>	May 2000	Pugaczewski et al.	
<u>6094531</u>	July 2000	Allison et al.	
<u>6175845</u>	January 2001	Smith et al.	
<u>6189116</u>	February 2001	Mongan et al.	
<u>6249882</u>	June 2001	Testardi	
<u>6259911</u>	July 2001	Bims et al.	455/423
<u>6272673</u>	August 2001	Dale et al.	
<u>6385594</u>	May 2002	Lebda et al.	
<u>6405364</u>	June 2002	Bowman-Amuah	717/101
<u>6408335</u>	June 2002	Schwaller et al.	
<u>6408403</u>	June 2002	Rodrigues et al.	714/38
<u>6421793</u>	July 2002	Lester et al.	
<u>6424978</u>	July 2002	Liu et al.	715/501.1
<u>6442748</u>	August 2002	Bowman-Amuah	
<u>6480469</u>	November 2002	Moore et al.	
<u>6502102</u>	December 2002	Haswell et al.	707/102

## OTHER PUBLICATIONS

Test Director (TM) 5: Integrated Test Management.; 1998 Mercury Interactive Corporation; p. 1-4.  
 Press Release, "Mt. Arlington, New Jersey, USA--Mar. 15, 1999"; p. 1-2.  
 Kasik et al., "Toward Automatic Generation of Novice User Test Scripts", ACM Press, 1996, p. 244-251.  
 Chays et al., "A Framework for Testing Database Applications", ACM, 2000, pp. 147-157.  
 Balcer et al.; Automatic Generation of Test Scripts from Formal Test Specifications.; ACM Press, IEEE-CS, SIGSOFT; 1989, pp. 210-218.  
 Memon et al., "Coverage Criteria for GUI Testing", ACM, 2001, pp. 256-267.

"LoadRunner (R) 6: The Enterprise Load Testing Tool"; 1999 Mercury Interactive Corporation.  
Banick et al.: "Web Management with Microsoft Visual SourceSafe 5.0"; Que Corporation; 1997.  
Nolan, C; " A look at e-Test Suite 3.1 by RSW"; Internet Citation, Jul. 1999, XP002155308.

ART-UNIT: 2122

PRIMARY-EXAMINER: Nguyen-Ba; Antony

ATTY-AGENT-FIRM: Oppenheimer Wolff & Donnelly LLP

ABSTRACT:

A system, method and article of manufacture are provided for affording test maintenance in an automated scripting framework. First, a plurality of test scripts are developed. Then, the plurality of test scripts are stored in a centrally located database. A user is then allowed to edit a specific test script located on the centrally located database. Finally, the user edits to the specific test script are propagated to each of the plurality of test scripts.

18 Claims, 82 Drawing figures

Full	Title	Citation	Front	Review	Classification	Date	Reference			Claims	KWIC	Draw De
------	-------	----------	-------	--------	----------------	------	-----------	--	--	--------	------	---------

☐ 6. Document ID: US 6633878 B1

L3: Entry 6 of 10

File: USPT

Oct 14, 2003

US-PAT-NO: 6633878

DOCUMENT-IDENTIFIER: US 6633878 B1

TITLE: Initializing an ecommerce database framework

DATE-ISSUED: October 14, 2003

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Underwood; Roy Aaron	Long Grove	IL		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Accenture LLP	Palo Alto	CA			02

APPL-NO: 09/ 364735 [PALM]

DATE FILED: July 30, 1999

INT-CL: [07] G06 F 17/30

US-CL-ISSUED: 707/100; 707/1, 707/102, 707/104, 707/205, 345/331

US-CL-CURRENT: 707/100; 707/1, 707/102, 707/205

FIELD-OF-SEARCH: 707/100, 707/1, 707/102, 707/104, 707/205, 707/2, 707/203,

345/331, 345/971, 345/329, 709/205, 709/224, 709/219, 709/202, 705/8, 711/113

PRIOR-ART-DISCLOSED:

## U.S. PATENT DOCUMENTS

PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
<u>5530829</u>	June 1996	Beardsley et al.	711/113
<u>6044368</u>	March 2000	Powers	707/2
<u>6112240</u>	August 2000	Pogue et al.	709/224
<u>6209000</u>	March 2001	Klien et al.	707/203
<u>6222535</u>	April 2001	Hurd, II	345/331

ART-UNIT: 2172

PRIMARY-EXAMINER: Shah; Sanjiv

ATTY-AGENT-FIRM: Burton; Daphne L. Oppenheimer Wolff &amp; Donnelly LLP

## ABSTRACT:

A system, method and article of manufacture are provided for initializing a database used with an issue tracker. The issue tracker receives information relating to a plurality of issues from a plurality of users, displays the information relating to the issues, and allows the browsing of the information relating to each of the issues. To initialize the database, the information relating to the issues is stored in a first database. A second database is also provided that stores tables including: a plurality of user interfaces; and/or application logic for accessing the information in the first database. The tables of the second database are reconfigured upon migrating the first database from a first folder to a second folder.

15 Claims, 179 Drawing figures

Full	Title	Citation	Front	Review	Classification	Date	Reference				Claims	KWC	Draw. D.
------	-------	----------	-------	--------	----------------	------	-----------	--	--	--	--------	-----	----------

☐ 7. Document ID: US 6609128 B1

L3: Entry 7 of 10

File: USPT

Aug 19, 2003

US-PAT-NO: 6609128

DOCUMENT-IDENTIFIER: US 6609128 B1

TITLE: Codes table framework design in an E-commerce architecture

DATE-ISSUED: August 19, 2003

## INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Underwood; Roy Aaron	Long Grove	IL		

## ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Accenture LLP	Palo Alto	CA			02

APPL-NO: 09/ 364491 [PALM]

DATE FILED: July 30, 1999

INT-CL: [07] G06 F 17/30

US-CL-ISSUED: 707/10; 707/104, 707/200

US-CL-CURRENT: 707/10; 707/200

FIELD-OF-SEARCH: 707/10, 707/1, 707/2, 707/101, 707/103, 707/104, 707/200, 341/51

PRIOR-ART-DISCLOSED:

## U.S. PATENT DOCUMENTS

PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
5844508	December 1998	Murashita et al.	341/51

ART-UNIT: 2172

PRIMARY-EXAMINER: Shah; Sanjiv

ATTY-AGENT-FIRM: Burton; Daphne L. Oppenheimer Wolff &amp; Donnelly LLP

## ABSTRACT:

A system, method and article of manufacture are provided for maintaining application consistency. First, a table of codes and associated text phrases are provided. Such table of codes is stored on a local storage medium within an e-commerce computer architecture. Next, the table of codes is accessed on the local storage medium within the e-commerce computer architecture. One of the text phrases is subsequently retrieved by selecting a corresponding one of the codes of the table. During operation, modification of the text phrases associated with each of the codes of the table is permitted. A plurality of services are executed, including retrieving a single one of the text phrases, retrieving all of the text phrases in response to a single command, updating a single code and text phrase combination, updating all of the code and text phrase combinations, naming the table, adding a new code and text phrase combination, removing one of the code and text phrase combination, and adding another table.

15 Claims, 179 Drawing figures

Full	Title	Citation	Front	Review	Classification	Date	Reference				Claims	KWIC	Draw. D.
------	-------	----------	-------	--------	----------------	------	-----------	--	--	--	--------	------	----------

☐ 8. Document ID: US 6601233 B1

L3: Entry 8 of 10

File: USPT

Jul 29, 2003

US-PAT-NO: 6601233



DOCUMENT-IDENTIFIER: US 6601233 B1

TITLE: Business components framework

DATE-ISSUED: July 29, 2003

## INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Underwood; Roy Aaron	Long Grove	IL		

## ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Accenture LLP	Palo Alto	CA			02

APPL-NO: 09/ 364533 [PALM]

DATE FILED: July 30, 1999

INT-CL: [07] G06 F 9/44

US-CL-ISSUED: 717/102; 717/100, 717/101, 717/103, 717/104, 717/106, 717/107

US-CL-CURRENT: 717/102; 717/100, 717/101, 717/103, 717/104, 717/106, 717/107

FIELD-OF-SEARCH: 717/1, 717/2, 717/3, 717/120, 717/100-108, 707/203, 707/104

PRIOR-ART-DISCLOSED:

## U.S. PATENT DOCUMENTS

PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
<u>5437027</u>	July 1995	Bannon et al.	
<u>5623601</u>	April 1997	Vu	
<u>5630131</u>	May 1997	Palevich et al.	
<u>5659735</u>	August 1997	Parrish et al.	
<u>5677997</u>	October 1997	Talatick	
<u>5754173</u>	May 1998	Hiura et al.	
<u>5764897</u>	June 1998	Khalidi	
<u>5819281</u>	October 1998	Cummins	
<u>5819295</u>	October 1998	Nakagawa et al.	
<u>5835911</u>	November 1998	Nakagawa et al.	
<u>5872973</u>	February 1999	Mitchell et al.	
<u>5956732</u>	September 1999	Tsuchida	
<u>5960200</u>	September 1999	Eager et al.	
<u>6014669</u>	January 2000	Slaughter et al.	
<u>6016495</u>	January 2000	McKeehan et al.	
<u>6029178</u>	February 2000	Martin et al.	
<u>6055538</u>	April 2000	Kessenich et al.	
<u>6108670</u>	August 2000	Weida et al.	
<u>6137869</u>	October 2000	Voit et al.	
<u>6167564</u>	December 2000	Fontana et al.	

<u>6182226</u>	January 2001	Reid et al.	
<u>6202051</u>	March 2001	Woolston	
<u>6230194</u>	May 2001	Frailong et al.	
<u>6233584</u>	May 2001	Purcell	
<u>6246410</u>	June 2001	Bergeron et al.	
<u>6256659</u>	July 2001	McLain et al.	
<u>6256678</u>	July 2001	Traugher et al.	
<u>6260068</u>	July 2001	Zalewski et al.	
<u>6282605</u>	August 2001	Moore	
<u>6286028</u>	September 2001	Cohen et al.	
<u>6304893</u>	October 2001	Gish	
<u>6308188</u>	October 2001	Bernardo et al.	
<u>6341287</u>	January 2002	Sziklai et al.	707/102
<u>6345239</u>	February 2002	Bowman-Amuah	703/6
<u>6370573</u>	April 2002	Bowman-Amuah	709/223
<u>6401085</u>	June 2002	Gershman et al.	
<u>6405364</u>	June 2002	Bowman-Amuah	717/101
<u>6430556</u>	August 2002	Goldberg et al.	
<u>6438514</u>	August 2002	Hill et al.	
<u>6442620</u>	August 2002	Thatte et al.	
<u>6473794</u>	October 2002	Guheen et al.	709/223

## OTHER PUBLICATIONS

Title: Categorization of business systems component, Author: W. Kozaczynski, ACM, 1996.\*

Title: Architecture framework for business components, author: Kozaczynski, W, IEEE, 1998.\*

Title: Business benefits from enterprise modeling and integration, author: Weston, R. H, IEEE, 1997.\*

"Sun HIPPI 1.0 User's Installation Guide", Mar. 1999, pp. 69-70 (Glossary).

<[sunsolve.sun.com/data/805/805-7133/pdf/014.glossary.pdf](http://sunsolve.sun.com/data/805/805-7133/pdf/014.glossary.pdf)>.

"Internet Firewalls Frequently Asked Questions" by Marcus J. Ranum and Matt Curtin, May 26, 1998. <<http://195.195.128.170/staff/scotty/firewallfaq/fwfaq.html>>.

ART-UNIT: 2762

PRIMARY-EXAMINER: Morse; Gregory

ASSISTANT-EXAMINER: Das; Chameli C.

ATTY-AGENT-FIRM: Oppenheim Wolff & Donnelly LLP

## ABSTRACT:

A method of generating software based on business components. A plurality of logical business components in a business are first defined with each business component having a plurality of capabilities. Next, functional interrelationships are identified between the logical business components. Code modules are then generated to carry out the capabilities of the logical business components and the functional interrelationships between the logical business components, wherein the code modules represent a transformation of the logical business components to their physical implementation, while ensuring the capabilities that are carried out by

each code module are essentially unique to the logical business component associated with the code module. Next, the functional aspects of the code modules and the functional relationships of the code modules are tested. The code modules are then subsequently deployed in an e-commerce environment.

18 Claims, 177 Drawing figures

Full	Title	Citation	Front	Review	Classification	Date	Reference			Claims	RWMC	Draw. De
------	-------	----------	-------	--------	----------------	------	-----------	--	--	--------	------	----------

☐ 9. Document ID: US 6523027 B1

L3: Entry 9 of 10

File: USPT

Feb 18, 2003

US-PAT-NO: 6523027

DOCUMENT-IDENTIFIER: US 6523027 B1

TITLE: Interfacing servers in a Java based e-commerce architecture

DATE-ISSUED: February 18, 2003

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Underwood; Roy Aaron	Long Grove	IL		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Accenture LLP	Palo Alto	CA			02

APPL-NO: 09/ 364531 [PALM]

DATE FILED: July 30, 1999

INT-CL: [07] G06 F 17/30

US-CL-ISSUED: 707/4; 707/10, 707/100

US-CL-CURRENT: 707/4; 707/10, 707/100

FIELD-OF-SEARCH: 707/4, 707/3, 707/2, 707/5, 707/6, 707/1, 707/8, 707/9, 707/10, 707/100, 707/102, 707/103, 709/205, 709/247, 709/250

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
<u>5758074</u>	May 1998	Marlin et al.	709/250
<u>6112228</u>	August 2000	Earl et al.	709/205
<u>6185625</u>	February 2001	Tso et al.	709/247
<u>6199068</u>	March 2001	Carpenter	707/100
<u>6199079</u>	March 2001	Gupta et al.	707/507
<u>6230160</u>	May 2001	Chan et al.	707/103
<u>6266666</u>	July 2001	Ireland et al.	707/10

## OTHER PUBLICATIONS

Betz M, "Active data Objects & ASP", Dr. Dobb's Journal, U.S. M&T Publ., May 1998, vol. 23, No. 5, pp. 88-91, 111-112.  
 Spitzer, "Tiers Without Tears", DBMS Online, Jan. 12, 1998, pp. 1-5.  
 Vinoski, "Corba: Integrating Diverse Applications Within Distributed Heterogeneous Environments", IEEE Communications Magazine, Feb. 1997, pp. 46-55.

ART-UNIT: 2172

PRIMARY-EXAMINER: Shah; Sanjiv

ATTY-AGENT-FIRM: Burton; Daphne L. Oppenheimer Wolff & Donnelly, LLP

## ABSTRACT:

A system, method and article of manufacture are provided for providing an interface between a first server and a second server with a proxy component situated therebetween. Initially, a request for a business object is identified by an application on the first server. The first server is then connected to the second server. Next, selection criteria from the first server is transmitted to the second server. In response to the selection criteria, the first server receives a first recordset and a second recordset from the second server. Business data is included in the first recordset and result codes are included in the second recordset. The first and second recordsets are mapped to the business object and the business object is sent to the application on the first server.

18 Claims, 179 Drawing figures

Full	Title	Citation	Front	Review	Classification	Date	Reference			Claims	RWC	Draw. D.
------	-------	----------	-------	--------	----------------	------	-----------	--	--	--------	-----	----------

☐ 10. Document ID: US 6502102 B1

L3: Entry 10 of 10

File: USPT

Dec 31, 2002

US-PAT-NO: 6502102

DOCUMENT-IDENTIFIER: US 6502102 B1

**\*\* See image for Certificate of Correction \*\***

TITLE: System, method and article of manufacture for a table-driven automated scripting architecture

DATE-ISSUED: December 31, 2002

## INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Haswell; John Jeffrey	Herndon	VA		
Young; Robert J.	Charlestown	MA		
Schramm; Kevin	Rose Valley	PA		

## ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Accenture LLP	Palo Alto	CA			02

APPL-NO: 09/ 536878 [PALM]  
 DATE FILED: March 27, 2000

INT-CL: [07] G06 F 17/30

US-CL-ISSUED: 707/102  
 US-CL-CURRENT: 707/102

FIELD-OF-SEARCH: 707/200, 707/102, 717/124, 703/22, 714/4, 714/38, 714/43, 714/47

PRIOR-ART-DISCLOSED:

# U.S. PATENT DOCUMENTS

PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
5905856	May 1999	Ottenssooser	714/38

ART-UNIT: 2175

PRIMARY-EXAMINER: Mizrahi; Diane D.

ASSISTANT-EXAMINER: Mofiz; Apu

ATTY-AGENT-FIRM: Oppenheimer Wolff & Donnelly LLP

## ABSTRACT:

A system, method and article of manufacture are provided for affording a table-driven automated scripting architecture. First, test script information is divided into a plurality of components of one or more words having a commonly understood meaning. Then the components are stored into a database. Later, the components are parsed into one or more words (each having a commonly understood meaning). The database is queried for the words to retrieve a set of computer instructions that cause a computer to perform functions related to the commonly understood meaning of the words and then to perform those functions.

20 Claims, 82 Drawing figures

Full	Title	Citation	Front	Review	Classification	Date	Reference			Claims	KWIC	Draw. De
------	-------	----------	-------	--------	----------------	------	-----------	--	--	--------	------	----------

Clear	Generate Collection	Print	Fwd Refs	Blwd Refs	Generate OACS
-------	---------------------	-------	----------	-----------	---------------

Term	Documents
EVENT	828889
EVENTS	194086
PROVIDER	69433
PROVIDERS	35628
(2 AND (PROVIDER NEAR EVENT)).PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD.	10

(L2 AND (EVENT NEAR PROVIDER)).PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD.
---

10
----

**Display Format:****Change Format**[Previous Page](#)[Next Page](#)[Go to Doc#](#)

[First Hit](#)   [Fwd Refs](#)

Generate Collection

Print

L5: Entry 8 of 28

File: USPT

Apr 13, 2004

DOCUMENT-IDENTIFIER: US 6721713 B1

TITLE: Business alliance identification in a web architecture framework

Detailed Description Text (73):

To date, Web development tools have been limited in their ability to create dynamic Web applications which span from client to server and interoperate with existing computing resources. Until recently, HTML has been the dominant technology used in development of Web-based solutions. However, HTML has proven to be inadequate in the following areas:

Detailed Description Text (84):

Sun's Java language has emerged as an industry-recognized language for "programming the Internet." Sun defines Java as: "a simple, object-oriented, distributed, interpreted, robust, secure, architecture-neutral, portable, high-performance, multithreaded, dynamic, buzzword-compliant, general-purpose programming language. Java supports programming for the Internet in the form of platform-independent Java applets." Java applets are small, specialized applications that comply with Sun's Java Application Programming Interface (API) allowing developers to add "interactive content" to Web documents (e.g., simple animations, page adornments, basic games, etc.). Applets execute within a Java-compatible browser (e.g., Netscape Navigator) by copying code from the server to client. From a language standpoint, Java's core feature set is based on C++. Sun's Java literature states that Java is basically, "C++ with extensions from Objective C for more dynamic method resolution."

Detailed Description Text (130):

FIG. 54 is an illustration showing a security organization according to one embodiment of the present invention. A Security Management Team may have a security management 300, under which are an administration team 302, a projects & planning team 304, and a business process security team 306. The size of the Security Management team, and the way in which it is integrated into the development organization depends on the degree to which security is a factor for each specific environment. For example, the security risks associated with an Internet-based online banking system are far greater than those of a fully isolated client/server system, and therefore warrant a larger team with broader responsibilities and greater influence.

Detailed Description Text (327):

It is important to set up and communicate a detailed folder structure with specified access rights from the beginning. Contents of folders must be checked regularly to ensure that folders contain what they are supposed to.

Detailed Description Text (332):

Another important distinction is the one between work in progress and completed documents that have been approved. This distinction can be supported by a folder structure with carefully chosen access rights.

Detailed Description Text (625):

Although direct sabotage is rare, inexperienced developers, perhaps new to the project, can wreak havoc to the system under development by inadvertently deleting

or modifying system components. Focus must be on defining access rights so that developers have the right level of access (read/write) to all the information that is useful and relevant to their work.

Detailed Description Text (806):

Assembly Test--The assembly test tests the interaction of related components to ensure that the components, when integrated, function properly. Assembly test ensures that data is passed correctly between screens in a conversation or batch process and that messages are passed correctly between a client and a server. The specification tested is the technical design. The application flow diagram within the technical design depicts the assemblies, either on-line conversations or batch assemblies, that will be assembly tested. Testing is therefore organized by assembly rather than by business function.

Detailed Description Text (820):

The Operational Readiness Test--The objective of the operational readiness test is to ensure that the application can be correctly deployed. The operational readiness test is also commonly known as the readiness test, roll-out test, release test, or the conversion test. The operational readiness test becomes especially key in client/server environments. It has four parts: Roll out test--ensures that the roll out procedures and programs can install the application in the production environment. Operations test--ensures that all operational procedures are in place and acceptable, and that the production system can be operated by the personnel responsible for supporting production. Service level test--ensures that once the application is rolled out, it provides the level of service to the users as specified in the Service Level Agreement (SLA). Roll out verification--ensures that the application has been correctly rolled out at each site. This test, developed by the work cell or team performing operational readiness test, should be executed during each site installation by the work cell or team in charge of the actual roll out of the application.

Detailed Description Text (948):

For any process where multiple groups are involved, well-defined procedures must be in place to ensure that work flows from one task to another. Each participant must have access to the information required to perform the task, including the information from previous steps in the flow. This can be handled manually or supported by tools. If handled manually, it requires dedication, attention to detail, and significant training.

Detailed Description Text (993):

Role-based access control establishes access rights and profiles based on job functions within the environment. If different access rights are required for security administrators vs. code developers vs. code reviewers vs. testers, then the correct access can be established based on these functions.

Detailed Description Text (1032):

Repository access control is important where developers in the development environment need to be assigned different rights to the repository. Typically, the developers will be placed in groups with diminishing access rights such as repository administrator, technical support, designer, or programmer. These access rights may relate to read/write/modify/delete authority. This method of access control is far more flexible than simple object locking.

Detailed Description Text (1071):

Repository access can sometimes be controlled using an access control function, which comes with the repository. A common technique is to group users and assign different access rights to the different groups. Each of these groups is also assigned specific read/write/delete/modify authority. For example, the following groups may be defined as having increasing rights:



Detailed Description Text (1096):

Flexible access rights based on user profiles, which differentiate (at least) between read and write access

Detailed Description Text (1191):

g) Does the tool provide ease of access to information?

Detailed Description Text (1334):

Security tools are required in the development environment to ensure against unauthorized access by individuals and system processes, to limit damages caused by such unauthorized access, and to audit access the environment services. At the security management level, it may be valuable to have tools which help manage security profiles, security groups, and access rights.

Detailed Description Text (1337):

Role-based access control establishes access rights and profiles based on job functions within the environment. If different access rights are required for security administrators vs. code developers vs. code reviewers vs. testers, then the correct access can be established based on these functions.

Detailed Description Text (1352):

Performance modeling tools in this category support the analysis of the development environment's performance, as opposed to that of the client/server application being developed. A simple spreadsheet may be suitable in some well-known and understood environments, but dedicated performance modeling tools should be considered on any project with high transaction volumes or complex environments involving multiple platforms.

Detailed Description Text (1383):

Test Data Management--Test results, expected results, and data comparison results can be linked to a defect to provide centralized access to the information. Integration also aids in keeping track of the cycle where the problem occurred, the test condition, and therefore the business function affected by the problem.

Detailed Description Text (1403):

Design tools are used to specify "how" a system will implement these system requirements. They are typically diagramming tools, which graphically depict how the system will be built in terms of its key components. This differs between classical client/server systems and component-based systems:

Detailed Description Text (1404):

The standard client/server model comprises application logic, presentation, and communication components, which together support the business processes. For a client/server system, each of these components must be individually defined.

Detailed Description Text (1461):

The information management component may provide the security needed in a multi-designer environment. If this is not the case then a multi-designer data modeling tool should be used. The tool may provide a central dictionary which allows design data to be shared between several designers and includes security checks to monitor any conflicts in overlapping access rights between designers.

Detailed Description Text (1535):

c) Are there hundreds of users? Are there tens of servers?

Detailed Description Text (1655):

Application Logic Design tools are used to graphically depict an application. These tools include application structure, module descriptions, and distribution of functions across client/server nodes.

Detailed Description Text (1656):

A variety of tools and techniques can be used for Application Logic Design. Examples are structure charts, procedure diagrams (module action diagrams), and graphics packages to illustrate distribution of functions across client and server.

Detailed Description Text (1779):

Communication design tools are essential in developing systems where critical business operations have to have maximum availability and minimum down time. One of the primary contributing factors to high performance in client/server environments is a good network design. A good network design can only be achieved through a good communication design.

Detailed Description Text (1804):

Reverse engineering tools are used to capture specific, relevant functional and design information from a legacy system for use in a new, client/server system or to restructure the existing system for improved performance and maintenance.

Detailed Description Text (1839):

Construction tools are used to program or build the application: client and server source code, windows, reports, and database. Along with the onset of Visual Programming, the more traditional form of construction tools have been superceded by Integrated Development Environments (IDEs) which take all the basic components required for construction, and integrate them into a single system. Although IDEs are now the preferred tools for most construction, the components that make up these tools remain the same--Source Code Editor, Compiler/Linker/Interpreter, Generation Tools and Debugging Tools.

Detailed Description Text (1840):

Visual Programming tools, initially associated with the rapid development of the client-side of client/server applications, have now matured and expanded their domain to cover entire client/server development (e.g. Visual C++) and Netcentric development (e.g. visual Java IDEs).

Detailed Description Text (1960):

Testing applications (client/server or Netcentric) remains a complex task because of the large number of integrated components involved (for example, multiplatform clients, multiplatform servers, multitiered applications, communications, distributed processing, and data), which, in turn, results in a large number and variety of Testing tools.

Detailed Description Text (2194):Event Management (1318)Detailed Description Text (2195):

An event is an electronic message generated by any component (e.g., application software, system software, hardware, etc.) in the system. Event Management receives, logs, classifies and presents event messages on a console(s) based on pre-established filters or thresholds.

Detailed Description Text (2238):

Automatic logging of problems will require interfaces to be built with the Event Management system, and perhaps the execution architecture for application errors.

Detailed Description Text (2252):

The way in which a disaster is defined will be dependent upon which resources are critical to the business. For example, a data center failure may be critical for one client whereas a server failure for another is more critical.

Detailed Description Text (2423):

Backups are typically embedded into production scheduling with restores on an ad hoc basis. Backup/Restore needs to ensure that a file can be only backed up/restored by users with the right access level. Furthermore, file transfer utilities need to be used when the information to archived is sent through the network as well as security for file control access and global authorization should be available and done in concert with the security management facility.

Detailed Description Text (2505):

Security must exist in various levels throughout the system in order to prevent unauthorized access. Security components must be packaged into a security architecture which can be effectively managed by an organization through their security management strategies. The Kerberos security approach within client/server architecture, for example, utilizes interconnected security servers and distributed security clients to provide security for the distributed environment.

Detailed Description Text (2675):

Capacity Planning & Modeling must coordinate the requirements across the system (e.g., networks, servers, workstations, CPU, etc.) Capacity is driven by the need to meet SLAs with the user communities and as part of the planning and modeling process, future threats to capacity should be identified.

Detailed Description Text (2762):

Managing hardware is all hardware directly used to manage the environment. This includes all staging components. These components are devoted to systems management functions. Examples of managing hardware include management servers, management controllers, management consoles, probes, and sniffers. One significant component in the hardware monitoring arena is Firewall access control policy management. Firewalls are regularly used for network based security management. It is typically a system or group of systems that enforce access control between two or more networks and/or perform network data packet filtering. Usually packet filtering router hardware and application gateways are used to block unauthorized IP packets and enforce proxy defined user commands.

Detailed Description Text (2783):

Event Management

Detailed Description Text (2784):

An event is an electronic message generated by any component (e.g., application software, system software, hardware, etc.) in the system. Event Management receives, logs, classifies and presents event messages on a console(s) based on pre-established filters or thresholds.

Detailed Description Text (2787):

The scope of events to be monitored will have a major impact on the approach taken for Event management and the tools selected.

Detailed Description Text (2795):

The number of events generated in the system will increase due to the complexity of the system. Devices will generate events as well as applications, the technical infrastructure, etc. Common event handling mechanisms will be required to provide management information in a simple, consistent format and to forward important events on for management purposes. In addition, filtering capabilities may also be needed at remote locations to prevent the streaming of events to central/master management consoles.

Detailed Description Text (2804):

The physical environment includes all the support indirectly involved in maintaining and managing the distributed environment. Initially it was thought client/server technology would make data centers obsolete. However, with the migration of mission critical processes to client/server environments, many servers

are being maintained in data centers in an effort to increase reliability. As a result, the importance of managing the physical environment has increased. Partially because it was initially believed not to be very important and because it does not relate directly to the information systems, the physical environment of the operational architecture is often overlooked. These systems include UPS, raised floor, power, site survey and preparation, wiring/cabling, climate control, etc.

Detailed Description Text (2836):

The Internet is a method of interconnecting physical networks and a set of conventions for using networks that allow the computers they reach to interact. Physically, the Internet is a huge, global network spanning over 92 countries and comprising 59,000 academic, commercial, government, and military networks, according to the Government Accounting Office (GAO), with these numbers expected to double each year. Furthermore, there are about 10 million host computers, 50 million users, and 76,000 World-Wide Web servers connected to the Internet. The backbone of the Internet consists of a series of high-speed communication links between major supercomputer sites and educational and research institutions within the U.S. and throughout the world.

Detailed Description Text (2992):

Any of the foregoing types of browsers may be employed to access various databases via the Internet in order to conduct electronic commerce-related business. Typical database or file-based shopping cart systems require that the user be uniquely identified in order to associate particular data stored on the server with a particular user. This requires the user to log-in or create an account, which is then stored in the server. Each subsequent request from the user must reference the unique identifier, either in the uniform resource locator (URL) or as hidden data passed back through a form submission. Either of these approaches require that the account or ID information of the user be stored on the remote server in the network for some definite period of time. Usually, the user must keep track of the account identifier in order that the prior session information can be retrieved.

Detailed Description Text (3014):

A new method of distributing and viewing information known as the World-Wide Web has recently become very popular on the global Internet. The World-Wide Web is a collection of servers connected to the Internet that provide multi-media information to users that request the information. The users access the information using client programs called "browsers" to display the multi-media information.

Detailed Description Text (3016):

To access the multi-media information available on World-Wide Web servers, a user runs a client browser program that accesses the HTML formatted documents stored on the HTTP servers connected to the global Internet. The client browser program retrieves the formatted information and provides the information in an appropriate manner to the user. For example, the client browser program displays graphical image information as images on the user's graphical display screen; plays video information as video animation on the user's graphical display screen; displays text information as text on the user's screen; and plays sound samples using the speakers on the user's computer system. "Mosaic", one popular client browser program, is widely available to the users of the global Internet.

Detailed Description Text (3017):

For a company that wishes to develop an online presence, creating a World-Wide Web Server would provide a feature rich online service available to customers and clients. A World-Wide Web Server can store images, text, animation, and sounds that provide information about the company. Furthermore, World-Wide Web Servers can be implemented on relatively simple computer systems, including personal computers.

Detailed Description Text (3018):

Most World-Wide Web Servers are coupled to the global Internet. By deploying a

World-Wide Web Server on the global Internet a company would create online service that is accessible to the millions of global Internet users.

Detailed Description Text (3019):

Alternatively, a company can deploy a HTTP server that is available to customers through dial-up phone service. A dial-up HTTP server would be accessible to customers and clients that do not have Internet access. Thus, by creating a simple HTTP server, any organization or corporation can create an online presence.

Detailed Description Text (3020):

However, quickly creating the HTML formatted documents required for a World-Wide Web Server is not a trivial task. Moreover, the standard HTTP server software, without any additional programming, is very limited. For example, without custom extensions, an HTTP server cannot accommodate complex transactions between a user and the HTTP server or integrate a database system into an online service. Although it is possible to write custom extensions to the HTTP server software using a conventional programming language, such custom extensions are difficult to write except by experienced programmers. Thus, to be able to quickly deploy full-featured HTTP servers, it would be desirable to have a development tool usable by non-programmers that allows a developer to quickly and easily create a full-featured online service based upon the HTTP and HTML standards.

Detailed Description Text (3027):

Four different types of commercial transactions might commonly occur in a commercial online service. First, a user may be charged for the right to access all or parts of a useful publicly accessible online system. Second, the online service may pay the user for performing some type of action such as winning a contest or completing a marketing survey. Third, an online service may charge a content provider for placing certain information on the online service. For example, a content provider can be charged for placing an advertisement on the online service. Finally, a content provider can be paid by the online service for providing information that users may wish to access, can be provided on a for-fee basis. Conversely, an online service provider may wish to pay third party content providers for placing useful material on the online service.

Detailed Description Text (3107):

Licensing schemes have adapted to the network environment as well as the individual personal computer. In a network environment, such as a client-server network, multiple users may access the same copy of a particular application. Consequently, the vendor can charge the network owner not for the number of copies installed on the network, but for the number of users having access to the software.

Detailed Description Text (3111):

If a license is not available, the client contacts another server to find the appropriate license. The client in the conventional system has the responsibility to obtain licenses from the various servers, and the individual servers provide resources at the client's request. To facilitate such licensing, the application typically includes a library of programs designed to contact the server, request a license, and track the resulting license.

Detailed Description Text (3113):

In addition, conventional licensing systems rely on code embedded in the application to establish the licensing attributes. Code is placed in the application which interprets information received from the server to establish licensing parameters. Because the behavior of the license is not established until after the request has been made and the license obtained, the user cannot read the license terms prior to the request. In addition, this system lacks flexibility. To change the licensing terms, the code in the application must be revised.

Detailed Description Text (3123):

When computer software products are used in a network environment (which may include computers running in various roles as workstations and servers of various types linked together over a data path), additional licensing challenges are present. For example, a network may permit a user at one node (which may be a terminal or workstation, for instance) to utilize a software product running at another node (which may be the network server or even another workstation). Consequently, the terms of the single-computer type of software license might not cover the usage of the software product on the network, or worse still (from the point of view of the licensor) might actually permit such a usage without additional compensation to the licensor. One approach to network licensing is to grant permission to use the program based on all of the nodes on the network, and to require a license for each node. Then typically the license fee may be increased as the number of nodes on the network increases. Another approach bases the license fee for a software product running on a network on the total number of individual users who might actually run the software, regardless of the number of nodes either on the network or running the software product at a given time. These approaches, however, have usually required the cooperation of the licensee, because additional nodes may be added to the network, or additional users may utilize the software, without the knowledge of the licensor, who is typically not present on the premises of the licensee. The licensor may reserve the right to audit the licensee's site, but such an audit is intrusive, expensive, and may alienate potential or actual customers for licenses. Although other approaches exist under which one might charge a single fee per server or per site or per entity, often on an individually negotiated basis, these approaches are often impractical or inflexible, in that they also typically do not take into account the possible wide variation over time in the number of nodes or users and also require reliance on licensee cooperation.

Detailed Description Text (3315):

Commercial content providers are concerned with ensuring proper compensation for the use of their electronic information. Electronic digital information, for example a CD recording, can today be copied relatively easily and inexpensively. Similarly, unauthorized copying and use of software programs deprives rightful owners of billions of dollars in annual revenue according to the International Intellectual Property Alliance. Content providers and distributors have devised a number of limited function rights protection mechanisms to protect their rights. Authorization passwords and protocols, license servers, "lock/unlock" distribution methods, and non-electronic contractual limitations imposed on users of shrink-wrapped software are a few of the more prevalent content protection schemes. In a commercial context, these efforts are inefficient and limited solutions.

Detailed Description Text (3333):

Distribution using WAF may package both the electronic content and control information into the same WAF container, and/or may involve the delivery to an end-user site of different pieces of the same WAF managed property from plural separate remote locations and/or in plural separate WAF content containers and/or employing plural different delivery means. Content control information may be partially or fully delivered separately from its associated content to a user WAF installation in one or more WAF administrative objects. Portions of said control information may be delivered from one or more sources. Control information may also be available for use by access from a user's WAF installation secure sub-system to one or more remote WAF secure sub-systems and/or WAF compatible, certified secure remote locations. WAF control processes such as metering, budgeting, decrypting and/or fingerprinting, may as relates to a certain user content usage activity, be performed in a user's local WAF installation secure subsystem, or said processes may be divided amongst plural secure subsystems which may be located in the same user WAF installations and/or in a network server and in the user installation. For example, a local WAF installation may perform decryption and save any, or all of, usage metering information related to content and/or electronic appliance usage at such user installation could be performed at the server employing secure (e.g., encrypted) communications between said secure subsystems. Said server location may

also be used for near real time, frequent, or more periodic secure receipt of content usage information from said user installation, with, for example, metered information being maintained only temporarily at a local user installation.

Detailed Description Text (3340):

Control information may be provided by a party who does not directly participate in the handling of electronic content (and/or appliance) and/or control information for such content (and/or appliance). Such control information may be provided in secure form using WAF installation secure sub-system managed communications (including, for example, authenticating the deliverer of at least in part encrypted control information) between such not directly participating one or more parties' WAF installation secure subsystems, and a pathway of WAF content control information participant's WAF installation secure subsystem. This control information may relate to, for example, the right to access credit supplied by a financial services provider, the enforcement of regulations or laws enacted by a government agency, or the requirements of a customer of WAF managed content usage information (reflecting usage of content by one or more parties other than such customer) relating to the creation, handling and/or manner of reporting of usage information received by such customer. Such control information may, for example, enforce societal requirements such as laws related to electronic commerce.

Detailed Description Text (3610):

The education related services component of the present invention educates users over a network such as a LAN, WAN, an intranet, the internet, etc. Note operation 2504 of FIG. 83. Courses may be taken live, directly from a server, or downloaded to the workstation of a user. Bulletins relating to the courses could be posted on electronic bulleting boards or electronically mailed to individual students. Tests may also be offered over individual courses as well as entire curriculum.

Detailed Description Text (3771):

Provides ability to grant varying levels of access based on user identity

Detailed Description Text (3772):

The security component of the present invention may also permit users to create secure virtual networks between their systems. One example would be two branches of a business in two different cities being connected by a virtual network. Guaranteed secure data transfer may be offered. Further, remote login is allowed. As an option, the ability to grant varying levels of access based on user identity may be granted. This could include both access to the virtual network, and to any individual resources shared through the network.

Detailed Description Text (3781):

Passes requests from external clients to internal web servers and returns results

Detailed Description Text (3786):

The network services component of the present invention passes requests from external clients to internal web servers and returns results. This component may also serve as trusted agent to access machines on the behalf of clients, particularly useful for automatic upgrades or information downloading for offline use. Optionally, IP Addresses of machines may be hidden from external clients. Further, configuration control may be provided over access permissions. As another option, reverse proxy services could be provided.

Detailed Description Text (3821):

Serves requested web pages and graphics from web servers to client web browsers

Detailed Description Text (3824):

The internet services component of the present invention serves requested web pages and graphics from web servers to client web browsers. Page rendering for multiple languages may be supported, as may transmission of data to and from multiple

content sources such as file systems, databases, and scripts.

Detailed Description Text (3854):

Supports Server Information with Client Cookies

Detailed Description Text (3856):

The internet services component of the present invention may provide a mechanism to note and remember one or more preceding events in a given sequence of interactions with the user or application program. State and session information may be tracked. Multiple independent user sessions that are simultaneously active may be managed. Support is provided for user cookies, server information with user cookies, client URL encoding, and server information with URL session identifiers.

Detailed Description Text (3966):

End-user multimedia may be delivered across all bandwidths. As an option, customizable web-based server administration and reporting could be offered to aid business management. Enhanced security would be used for sensitive or pay-per-view content. Ideally, multimedia capabilities would integrate with existing back office applications. Customized applications and leveraged content could be created in existing formats. Also offered could be the ability to scale with additional hardware. Ideally, the multimedia components support multiple concurrent users.

Detailed Description Text (4053):

The management and operations component of the present invention centrally creates and manages policies and user profiles. Hardware inventories for workstations may also be automatically centrally created. Printers are centrally configured and loads are balanced. Centralized application installation may be offered. As an option, particular application rights may be assigned to groups or individuals and a standard desktop environment may be offered to some or all users. Ideally, automatic transparent transfer across multiple servers is permitted. Optionally, remote help desk utilities for software-related problems are provided.

Detailed Description Text (4100):

Allows user to set various access levels to assign user- and project-specific authorization

Detailed Description Text (4101):

Allows user to tailor custom file access rights

Detailed Description Text (4106):

The web development component of the present invention may store current files along with past changes to docs to allow easy re-creation of previous versions. Users may be allowed to set various access levels to assign user- and project-specific authorization. Preferably, users are allowed to tailor custom file access rights.

Detailed Description Text (4136):

employ "templates" to ease the process of configuring capabilities of the present invention as they relate to specific industries or businesses. Templates are applications or application add-ons under the present invention. Templates support the efficient specification and/or manipulation of criteria related to specific content types, distribution approaches, pricing mechanisms, user interactions with content and/or administrative activities, and/or the like. Given the very large range of capabilities and configurations supported by the present invention, reducing the range of configuration opportunities to a manageable subset particularly appropriate for a given business model allows the full configurable power of the present invention to be easily employed by "typical" users who would be otherwise burdened with complex programming and/or configuration design responsibilities template applications can also help ensure that WAF related processes are secure and optimally bug free by reducing the risks associated with



the contribution of independently developed load modules, including unpredictable aspects of code interaction between independent modules and applications, as well as security risks associated with possible presence of viruses in such modules. WAF, through the use of templates, reduces typical user configuration responsibilities to an appropriately focused set of activities including selection of method types (e.g. functionality) through menu choices such as multiple choice, icon selection, and/or prompting for method parameter data (such as identification information, prices, budget limits, dates, periods of time, access rights to specific content, etc.) that supply appropriate and/or necessary data for control information purposes. By limiting the typical (non-programming) user to a limited subset of configuration activities whose general configuration environment (template) has been preset to reflect general requirements corresponding to that user, or a content or other business model can very substantially limit difficulties associated with content containerization (including placing initial control information on content), distribution, client administration, electronic agreement implementation, end-user interaction, and clearinghouse activities, including associated interoperability problems (such as conflicts resulting from security, operating system, and/or certification incompatibilities). Use of appropriate WAF templates can assure users that their activities related to content WAF containerization, contribution of other control information, communications, encryption techniques and/or keys, etc. will be in compliance with specifications for their distributed WAF arrangement. WAF templates constitute preset configurations that can normally be reconfigurable to allow for new and/or modified templates that reflect adaptation into new industries as they evolve or to reflect the evolution or other change of an existing industry. For example, the template concept may be used to provide individual, overall frameworks for organizations and individuals that create, modify, market, distribute, consume, and/or otherwise use movies, audio recordings and live performances, magazines, telephony based retail sales, catalogs, computer software, information data bases, multimedia, commercial communications, advertisements, market surveys, infomercials, games, CAD/CAM services for numerically controlled machines, and the like. As the context surrounding these templates changes or evolves, template applications provided under the present invention may be modified to meet these changes for broad use, or for more focused activities. A given WAF participant may have a plurality of templates available for different tasks. A party that places content in its initial WAF container may have a variety of different, configurable templates depending on the type of content and/or business model related to the content. An end-user may have different configurable templates that can be applied to different document types (e-mail, secure internal documents, database records, etc.) and/or subsets of users (applying differing general sets of control information to different bodies of users, for example, selecting a list of users who may, under certain preset criteria, use a certain document). Of course, templates may, under certain circumstances have fixed control information and not provide for user selections or parameter data entry.

Detailed Description Text (4144):

enable a user to securely extract, through the use of the secure subsystem at the user's WAF installation, at least a portion of the content included within a WAF content container to produce a new, secure object (content container), such that the extracted information is maintained in a continually secure manner through the extraction process. Formation of the new WAF container containing such extracted content shall result in control information consistent with, or specified by, the source WAF content container, and/or local WAF installation secure subsystem as appropriate, content control information. Relevant control information, such as security and administrative information, derived, at least in part, from the parent (source) object's control information, will normally be automatically inserted into a new WAF content container object containing extracted WAF content. This process typically occurs under the control framework of a parent object and/or WAF installation control information executing at the user's WAF installation secure subsystem (with, for example, at least a portion of this inserted control

information being stored securely in encrypted form in one or more permissions records). In an alternative embodiment, the derived content control information applied to extracted content may be in part or whole derived from, or employ, content control information stored remotely from the WAF installation that performed the secure extraction such as at a remote server location. As with the content control information for most WAF managed content, features of the present invention allows the content's control information to: (a) "evolve," for example, the extractor of content may add new control methods and/or modify control parameter data, such as WAF application compliant methods, to the extent allowed by the content's in-place control information. Such new control information might specify, for example, who may use at least a portion of the new object, and/or how said at least a portion of said extracted content may be used (e.g. when at least a portion may be used, or what portion or quantity of portions may be used); (b) allow a user to combine additional content with at least a portion of said extracted content, such as material authored by the extractor and/or content (for example, images, video, audio, and/or text) extracted from one or more other WAF container objects for placement directly into the new container; (c) allow a user to securely edit at least a portion of said content while maintaining said content in a secure form within said WAF content container; (d) append extracted content to a pre-existing WAF content container object and attach associated control information--in these cases, user added information may be secured, e.g., encrypted, in part or as a whole, and may be subject to usage and/or auditing control information that differs from the those applied to previously in place object content; (e) preserve WAF control over one or more portions of extracted content after various forms of usage of said portions, for example, maintain content in securely stored form while allowing "temporary" on screen display of content or allowing a software program to be maintained in secure form but transiently decrypt any encrypted executing portion of said program (all, or only a portion, of said program may be encrypted to secure the program).

Detailed Description Text (4147):

enable flexible metering of, or other collection of information related to, use of electronic content and/or electronic appliances. A feature of the present invention enables such flexibility of metering control mechanisms to accommodate a simultaneous, broad array of: (a) different parameters related to electronic information content use; (b) different increment units (bytes, documents, properties, paragraphs, images, etc.) and/or other organizations of such electronic content; and/or (c) different categories of user and/or WAF installation types, such as client organizations, departments, projects, networks, and/or individual users, etc. This feature of the present invention can be employed for content security, usage analysis (for example, market surveying), and/or compensation based upon the use and/or exposure to WAF managed content. Such metering is a flexible basis for ensuring payment for content royalties, licensing, purchasing, and/or advertising. A feature of the present invention provides for payment means supporting flexible electronic currency and credit mechanisms, including the ability to securely maintain audit trails reflecting information related to use of such currency or credit. WAF supports multiple differing hierarchies of client organization control information wherein an organization client administrator distributes control information specifying the usage rights of departments, users, and/or projects. Likewise, a department (division) network manager can function as a distributor (budgets, access rights, etc.) for department networks, projects, and/or users, etc.

Detailed Description Text (4148):

provide scalable, integratable, standardized control means for use on electronic appliances ranging from inexpensive consumer (for example, television set-top appliances) and professional devices (and hand-held PDAs) to servers, mainframes, communication switches, etc. The scalable transaction management/auditing technology of the present invention will result in more efficient and reliable interoperability amongst devices functioning in electronic commerce and/or data

security environments. As standardized physical containers have become essential to the shipping of physical goods around the world, allowing these physical containers to universally "fit" unloading equipment, efficiently use truck and train space, and accommodate known arrays of objects (for example, boxes) in an efficient manner, so WAF electronic content containers may, as provided by the present invention, be able to efficiently move electronic information content (such as commercially published properties, electronic currency and credit, and content audit information), and associated content control information, around the world. Interoperability is fundamental to efficient electronic commerce. The design of the WAF foundation, WAF load modules, and WAF containers, are important features that enable the WAF node operating environment to be compatible with a very broad range of electronic appliances. The ability, for example, for control methods based on load modules to execute in very "small" and inexpensive secure sub-system environments, such as environments with very little read/write memory, while also being able to execute in large memory sub-systems that may be used in more expensive electronic appliances, supports consistency across many machines. This consistent WAF operating environment, including its control structures and container architecture, enables the use of standardized WAF content containers across a broad range of device types and host operating environments. Since WAF capabilities can be seamlessly integrated as extensions, additions, and/or modifications to fundamental capabilities of electronic appliances and host operating systems, WAF containers, content control information, and the WAF foundation will be able to work with many device types and these device types will be able to consistently and efficiently interpret and enforce WAF control information. Through this integration users can also benefit from a transparent interaction with many of the capabilities of WAF. WAF integration with software operating on a host electronic appliance supports a variety of capabilities that would be unavailable or less secure without such integration. Through integration with one or more device applications and/or device operating environments, many capabilities of the present invention can be presented as inherent capabilities of a given electronic appliance, operating system, or appliance application. For example, features of the present invention include: (a) WAF system software to in part extend and/or modify host operating systems such that they possess WAF capabilities, such as enabling secure transaction processing and electronic information storage; (b) one or more application programs that in part represent tools associated with WAF operation; and/or (c) code to be integrated into application programs, wherein such code incorporates references into WAF system software to integrate WAF capabilities and makes such applications WAF aware (for example, word processors, database retrieval applications, spreadsheets, multimedia presentation authoring tools, film editing software, music editing software such as MIDI applications and the like, robotics control systems such as those associated with CAD/CAM environments and NCM software and the like, electronic mail systems, teleconferencing software, and other data authoring, creating, handling, and/or usage applications including combinations of the above). These one or more features (which may also be implemented in firmware or hardware) may be employed in conjunction with a WAF node secure hardware processing capability, such as a microcontroller(s), microprocessor(s), other CPU(s) or other digital processing logic.

Detailed Description Text (4159):

support smart card implementations of the present invention in the form of portable electronic appliances, including cards that can be employed as secure credit, banking, and/or money cards. A feature of the present invention is the use of portable WAFs as transaction cards at retail and other establishments, wherein such cards can "dock" with an establishment terminal that has a WAF secure sub-system and/or an online connection to a WAF secure and/or otherwise secure and compatible subsystem, such as a "trusted" financial clearinghouse (e.g., VISA, Mastercard). The WAF card and the terminal (and/or online connection) can securely exchange information related to a transaction, with credit and/or electronic currency being transferred to a merchant and/or clearinghouse and transaction information flowing

back to the card. Such a card can be used for transaction activities of all sorts. A docking station, such as a PCMCIA connector on an electronic appliance, such as a personal computer, can receive a consumer's WAF card at home. Such a station/card combination can be used for on-line transactions in the same manner as a WAF installation that is permanently installed in such an electronic appliance. The card can be used as an "electronic wallet" and contain electronic currency as well as credit provided by a clearinghouse. The card can act as a convergence point for financial activities of a consumer regarding many, if not all, merchant, banking, and on-line financial transactions, including supporting home banking activities. A consumer can receive his paycheck and/or investment earnings and/or "authentic" WAF content container secured detailed information on such receipts, through on-line connections. A user can send digital currency to another party with a WAF arrangement, including giving away such currency. A WAF card can retain details of transactions in a highly secure and database organized fashion so that financially related information is both consolidated and very easily retrieved and/or analyzed. Because of the WAF security, including use of effective encryption, authentication, digital signaturing, and secure database structures, the records contained within a WAF card arrangement may be accepted as valid transaction records for government and/or corporate recordkeeping requirements. In some embodiments of the present invention a WAF card may employ docking station and/or electronic appliance storage means and/or share other WAF arrangement means local to said appliance and/or available across a network, to augment the information storage capacity of the WAF card, by for example, storing dated, and/or archived, backup information. Taxes relating to some or all of an individual's financial activities may be automatically computed based on "authentic" information securely stored and available to said WAF card. Said information may be stored in said card, in said docking station, in an associated electronic appliance, and/or other device operatively attached thereto, and/or remotely, such as at a remote server site. A card's data, e.g. transaction history, can be backed up to an individual's personal computer or other electronic appliance and such an appliance may have an integrated WAF installation of its own. A current transaction, recent transactions (for redundancy), or all or other selected card data may be backed up to a remote backup repository, such a WAF compatible repository at a financial clearinghouse, during each or periodic docking for a financial transaction and/or information communication such as a user/merchant transaction. Backing up at least the current transaction during a connection with another party's WAF installation (for example a WAF installation that is also on a financial or general purpose electronic network), by posting transaction information to a remote clearinghouse and/or bank, can ensure that sufficient backup is conducted to enable complete reconstruction of WAF card internal information in the event of a card failure or loss.

Detailed Description Text (4162):

support, complete, modular separation of the control structures related to (1) content event triggering, (2) auditing, (3) budgeting (including specifying no right of use or unlimited right of use), (4) billing, and (5) user identity (WAF installation, client name, department, network, and/or user, etc.). The independence of these WAF control structures provides a flexible system which allows plural relationships between two or more of these structures, for example, the ability to associate a financial budget with different event trigger structures (that are put in place to enable controlling content based on its logical portions). Without such separation between these basic WAF capabilities, it would be more difficult to efficiently maintain separate metering, budgeting, identification, and/or billing activities which involve the same, differing (including overlapping), or entirely different, portions of content for metering, billing, budgeting, and user identification, for example, paying fees associated with usage of content, performing home banking, managing advertising services, etc. WAF modular separation of these basic capabilities supports the programming of plural, "arbitrary" relationships between one or differing content portions (and/or portion units) and budgeting, auditing, and/or billing control information. For example, under WAF, a budget limit of \$200 dollars or 300 German Marks a month may

be enforced for decryption of a certain database and 2 U.S. Dollars or 3 German Marks may be charged for each record of said database decrypted (depending on user selected currency). Such usage can be metered while an additional audit for user profile purposes can be prepared recording the identity of each filed displayed. Additionally, further metering can be conducted regarding the number of said database bytes that have been decrypted, and a related security budget may prevent the decrypting of more than 5% of the total bytes of said database per year. The user may also, under WAF (if allowed by senior control information), collect audit information reflecting usage of database fields by different individuals and client organization departments and ensure that differing rights of access and differing budgets limiting database usage can be applied to these client individuals and groups. Enabling content providers and users to practically employ such diverse sets of user identification, metering, budgeting, and billing control information results, in part, from the use of such independent control capabilities. As a result, WAF can support great configurability in creation of plural control models applied to the same electronic property and the same and/or plural control models applied to differing or entirely different content models (for example, home banking versus electronic shopping).

Detailed Description Paragraph Table (1):

Product Functionality Product Name/ Category Product Details Application A platform for the development, delivery and Server management of enterprise network applications. Based on CORBA and JAVA, Product1 uses an open and secure architecture to develop business applications. The Product1 product family consists of the following components: Product1 Studio - a visual integrated development environment tool for developing Java-based applications in Product1 and Java. It incorporates wizards and editors for creating web-based applications, including construction of user interface, data access and PACs. It also integrates with source code control, testing and deployment tools. Product1 Application Server - a Java-and CORBA- based server that provides state and session management, built-in load balancing, processing of application logic and integration with external databases and enterprise systems. Product1 Java Object Framework - a framework of reusable Java and JavaBeans objects. A host of Product1 Java classes and methods are available out-of-the-box for custom development. Product1 Command Center - a Java-based application that provides local and remote management and monitoring of the platform in real-time. This management console provides control of the application server, with the ability to configure a range of properties for each server component and the processes within them. It can also distribute components across multiple systems and manage multiple configurations. The Product1 product family may be extended through these components: PAC SDK - Product1 platform that allows developers to build customized Platform Adapter Components (PACs) for external enterprise systems. PACs - Business1 provides a PAC for SAP and PeopleSoft. Business1 partners deliver other 3rd party PACs that can be purchased from partners directly. Internet A family of Internet mail server products Mail that securely handles mail messages in a Server variety of formats. SIMS also provides a (SIMS) secure JAVA Administration Console for centralized and remote administration, backup and restore features. SIMS is a replacement for the UNIX sendmail program which has been the target of frequent system break-ins. Internet Targeted for internet service providers, News the Internet News Server is a full-featured news Server server which offers user-focused interfaces, streamed feeder/reader design, web-based installation and administration and remote access. The Internet News Server is a component of the Product2 ISP Server suite. Forum Workgroup collaboration tools that allow users to communicate in a heterogeneous environment of Business1 workstations, PCs and Macintosh computers. Forum allows users to share a whiteboard and applications with others and seamlessly transfer files and "chat" with co-workers. Personal Personal WebAccess - a customizable, compact web WebAccess browser for devices that run the PersonalJava Browser platform. Personal Web Access is designed for manufacturers who want to provide consumers with an easy way to access the Web and retrieve information from a variety of consumer devices, including screen phones, set-top boxes, and wireless

hand-held devices. The browser supports common internet services such as authentication, FTP, applets, audio and media files. HotJava HotJava Browser - a lightweight, customizable Browser browser designed for OEMs and developers who create web-enabled devices and applications. Product3 A secure, standard-based web server for accessing, managing, and distributing information over the Internet, extranets, or intranets. Product3 supports Java servlet development and network caching of web pages. Product3 simplifies management of website environments through delegation of administrative privileges such as access rights to administer meta-data components or load-balancing. Java The first commercially available Java service Web based on the JavaServer API framework for Java Server servlets. It uses servlet technology to enable server-side Java applications and provides access control and security features. Java Web Server provides session tracking that provides a mechanism to track how people use and navigate websites. It also provides remote administration and logging features. Directory A multi-protocol, scalable global directory for Server storing information such as user definitions, user profiles, network resource definitions, and configuration parameters. It employs naming, directory, and authentication protocols on top of a shared, distributed, object repository. Users and applications can use the directory to locate and access information from anywhere in the network. JavaWallet Java Electronic Commerce Framework (JECF) is Business1's new initiative to create a standard, secure framework within which to conduct business transactions using any combination of currencies and payment instruments such as credit and debit cards, electronic cash and checks, and small cards. The initial component of the JECF is the JavaWallet, a client-side application that will be distributed as a core component of the Java environment. JavaWallet will allow users of any Java-enabled web browser or operating system to purchase goods and services from JECF-compliant merchant websites. JavaWallet provides a single user interface for electronic transactions, secure from tampering. When a consumer uses a Java-enabled browser to navigate an online mall, selects goods and services for purchase, he can access the JavaWallet for home banking and portfolio management. The consumer owns the JavaWallet that will be used to complete purchases and banking transactions. The user may set spending limits and can monitor spending through an auditable transaction log. Privacy of all data is protected through the use of encryption and digital signatures. Merchants offer good and services for sale on the Internet using applets which adhere to the JavaWallet architecture. These applets may include interfaces to payment processing, security services, customer profile services and database services. The Java Wallet family consists of the following components: Java Commerce Business (JCC) - a client side solution for eCommerce transactions. JCC provides users with a wallet-like user interface, a database, and a platform that enables a variety of payment instruments and protocols. Commerce JavaBeans - enables developers to write components to extend JCC functionality such as interfacing with payment servers and other transaction protocols. Gateway Security Model - allows a secure shield around protected APIs and components. Java A card that is embedded with either a micro- Card processor and a memory chip or only a memory chip with non-programmable logic. The microprocessor card can add, delete, and otherwise manipulate information on the card, while a memory-chip card can only undertake a pre-defined operation. echeck A server that allows the use of electronic checks Server for transactions. Business1 echeck server verifies digital signatures, processes checks according to the business rules of the bank (e.g. a check over \$25,000 requires two signatures), returns invalid checks, and settles all valid checks. Product4 A range of security-based hardware and software Product that offers packet filtering, encryption, security Suite administration, virtual private network and access restriction. The Product4 Product Suite includes the following components: Product4 Secure Net - a complete set of products designed to establish perimeter defense, secure intranets, secure remote access, and secure extranets including the following: Product4 EFS - firewall and security server software that screens network traffic as defined by the organization's security policy. It also acts as a high-speed encryption server to protect information going over untrusted networks. Product4 SPF-200 - security platform for perimeter defense and electronic commerce. It provides stealthing to help protect an organization



from Internet attacks. Product4 SKIP - provides encryption and key management capabilities which enables PCs, workstations, and servers to achieve secure/authenticated communication. Business1.net A remote-access strategy and technology that enables users to securely access all personalized data, application and information from Java-enabled browsers. Business1.net uses recently acquired i-Planet's secure, remote access software. Calendar Designed to manage large-scale enterprise Server calendaring systems, Business1's Calendar Server is integrated with Business1 Internet Mail Server and provides the following features: Maintenance of Personal Calendars Group Scheduling Calendar Security Product5 A web server package solution that includes third- Internet party Internet and security products including the Server following: Software Product5 Administration Software - provides server Bundle setup, configuration, and management capabilities through a browser. The Product5 Internet Server can be administered remotely for user access control, email management, software installation and backup and recovery. Checkpoint FireWall-First! - firewall and security software that protects data and network from unauthorized access from the public Internet. It also offers packet-level filtering. Trend Interscan Virus Wall - virus scanning software that verifies and filters out viruses in communications such as files and emails that interact with the Product5 Internet Server. Business1 Internet Mail Server - a family of Internet mail server products that securely handles mail messages in a variety of formats. Network Associates WebStalkers-First Intrusion Detection - software that provides around-the- clock monitoring and response to intrusions and misuses of a site and its files. Business2 SuiteSpot Server including Business2's Calendar, Chat, Enterprise, Messaging and Directory Servers, LiveWire Pro and Informix database. Product2 Targeted for internet service providers, ISP Business1's Product2 ISP Server provides users Serv with a bundle of platform extensions including Bundle the following: Internet Administrator - provides secure, remote management of distributed ISP services Inter Services Monitor - monitors Internet services, identifies and manages network problems Directory Services - provides a multi-protocol, global directory for storing information Host Configuration - provides ISP host configuration features including quick, repeatable installation, Product2 security configuration, intrusion detection, server process monitoring, and log file management. Product4 SKIP - provides encryption and key management capabilities which enables PCs, workstations, and servers to achieve secure/authenticated communication Network Product2 Bandwidth Manager - a software product Management that enables efficient network resource management. Tools By preventing a small number of applications or users from consuming all available bandwidth, it ensures the quality of service to users and network

#### Detailed Description Paragraph Table (2):

availability to applications. Product6 Enterprise Manager - Business1's distributed network management foundation that manages large heterogeneous networks. Product6 Enterprise Manager supports and manages Java applications built for various network types. Product6 Site Manager & Product6 Domain Manager - offer centralized management for networks of up to 100 nodes. Product features include the following: Monitoring of events and network health for multiple local and remote environments Distribution of management data Management of file systems, print queues and user groups Balancing of management processing loads across the network Development Business1 offers a variety of development and testing tools including the following: Testing Development Tools: Tools EmbeddedJava Application Environment JavaBeans Development Kit JavaBlend Java Compiler Compiler Java Development Kit Java Dynamic Management Kit (JDMK) JavaHelp Java Management API (JMAPI) Java JIT Compiler Java SDK Java WorkShop NEOWorks Personal Java Application Environment Servlet Development Kit Product6 ASN.1 Compiler Business1 Performance Workshop Fortran Business1 Visual WorkShop C++ Business1 Workshop Teamware Testing Tools: JavaCheck Java Heap Analysis Tool JavaPureCheck JavaScope JavaSpec JavaStar JavaLoad System JavaPC Software - provides central administration Management and support for the Java platform on PC-based thin Tools client devices. JavaPC is targeted at OEMs designing thin-client devices such as transaction terminals, cash registers, kiosks and ATMs. Product2 Management Console - Java-based utility that

provides views of servers on the network and application on those servers. It allows administrators to add users, hosts or applications from any client on the network. Product6 Backup - provides automated, backup, recovery and storage management services for files and applications in a wide array of systems on the network including UNIX, NetWare, Windows NT, PC or Apple Macintosh systems. It also provides centralized administration and control through a unified view. Product6 AdminSuite - suite of tools for administering distributed systems and managing user accounts, hosts, groups, administrative data, printer, file system, disk and serial ports. Product5 j Software - browser-based graphical administration tool that provides centralized administration of JavaStation network computers and Java Webtops on PCs. Product5 j provides Java technology clients with connectivity to legacy databases and applications. Business1 Product7 - host-based software used to monitor and administer tape libraries via a Java-enabled Web browser. The Library Monitor allows event logging and notification, remote diagnostics, remote configuration, and remote monitoring of library activity and status.

#### Detailed Description Paragraph Table (3):

Product Name/ Directory Product Details Business2 A suite of pre-built applications that run on Commerce Business2's Application Server. These applications Product1 include buying, selling, merchandising, and delivering content over the Internet: ECProduct1 - Software for the integration of eCommerce applications with legacy systems. It provides for the sending, receiving, and encrypted transmission of documents among the heterogeneous systems of trading partners over the Internet. SellerProduct1 - An application designed to support advanced business-to-business selling over the Internet. SellerProduct1 allows for the enforcement of trading partner agreements and business rules. SellerProduct1 provides the capability to create company-specific catalogs which can be set up to present different products to different users based upon purchase eligibility. SellerProduct1 includes search features, management tools, and order management (including tax, shipping, and payment services.) BuyerProduct1 - An Internet-based corporate procurement application that automates order and delivery, supports complex trading relationships, and allows for the exchange of information via EDI or the Internet. PublishingProduct1 - An application that utilizes both passive and active customer profiling capabilities to create targeted advertising, and to deliver personalized information for superior customer service. Content management tools are combined with application development tools to allow to host and deploy multiple sites. MerchantProduct1 - An online business-to-consumer merchandising solution that provides the following features: A single shopping cart for each customer, forms filled with predefined account information, tax calculation and discounts, product availability, and up-to-date order status information. Payment systems, catalog creation and administration tools, an order management system, and rapid customization of site's business processes through modifiable business rules and presentation templates. Search capabilities, including hierarchical menus, parametric searches by attribute, and simple keyword searches. BillerProduct1 - An Internet bill presentment and payment (IBPP) solution, particularly for the banking and telecommunications industries. TradingProduct1 - A commerce exchange application that enables trading partners of varying size and technical sophistication to transact business over the Internet through in-context document turnaround capabilities, and customizable prepackaged forms. Business2 A comprehensive set of components that integrates Product browsing, email, web-based word processing, chat, and group scheduling to allow users to communicate, share, and access information. Business2 Product2 includes: Product3 - web browser with support for Java, JavaScript, and SSL Product4 - an Internet mail client. Product5 - a web authoring tool. Instant Product4 - enables people to communicate easily and privately in real time over an intranet or the Internet, either one-on-one or in a group. Calendar - delivers group scheduling based on a scalable real-time architecture. Browser Customization Business2 Business Customization Kit - enables Internet service providers, Internet content providers, hardware OEMs, and others to create customized version of Product2. Business2 Mission Control Desktop - cross-platform administration tools to configure, deploy, centrally manage, and update



Business2 Product2. Business2 A high-performance, scalable, web server software Enterprise for deploying the largest-scale web sites. Server Business2 Enterprise Server includes a built-in search engine and supports standard security and authentication. The integrated LiveWire Pro software also adds content management, data access, and session management capabilities. Business2 also offers FastTrack Server - an entry-level enterprise server with limited functionality. Business2 A middleware infrastructure that supports the Application development and deployment of transactional, Server business-critical Internet applications. Business2 Application Server operates with other Business2 products and includes the following two development tools: Application Builder - provides an integrated and productive web development environment that enables developers to rapidly deliver enterprise-class web applications. Extension Builder - allows corporations to develop custom integration with heterogeneous systems and applications across the enterprise. Business2 A directory server that acts as the central Directory repository for customer, supplier and employee Server information. Business2 Directory Server enables the integration, storage and management of directory information from disparate data sources. It also provides security, authentication and replication features. A Directory Software Developer's Kit provides application programming interfaces that enable developers to directory-enable their applications. Business2 A system for caching and filtering web content, Proxy log analysis, and boosting network performance. Server Business2 A calendar server that supports the scheduling Calend of meetings, appointments, and resources for Server thousands of users. Business2 A newsgroup server that provides collaboration Chat services through discussion groups. Business2 Server Chat Server also supports the moderation of content and administration of discussion groups. Business2 An email server that delivers messages with Messaging embedded sound, graphics, video files, HTML Server forms, Java applets, and desktop applications. Other Business2 sells a range of products that provide Directory a user and security management infrastructure & Security for large-scale eCommerce, extranet, and intranet Products applications. Business2 Certificate Management System - issues and manages digital certificates for extranet and e-commerce applications. Business2 Directory for Secure E-Commerce - expands the capabilities of Business2 Directory Server to provide additional flexibility of user and security administration for large-scale commerce and extranet applications. Business2 Delegated Administrator - provides customizable self-service administration for customers and partners to manage their own user and account information. Business2 Meta-Directory - enables Business2 Directory Server to be automatically synchronized with relational databases as well as network operating system, messaging, and enterprise resource planning system directories Business2 Security Services - enables developers to incorporate standard Internet security technologies into applications. Other Process Manager - Enables enterprises to automate Business2 and modify business processes such as contract Products negotiation, bidding and contractor management. Business2 Process Manager supports the development and deployment of processes across extranets and intranets, and manages them for overall efficiency and precision. Process Manager has four components: Business2 Process Manager Builder - a visual design environment for designing business processes using intuitive feature such as drag-and-drop functionality and pick lists. Processes may be stored in Business2's Directory Server. Business2 Process Manager Engine - the server-based engine that hosts processes designed with PM Builder. Business2 Process Manager Express - browser-based user interface to Process Manager business processes. Business2 Process Manager Administrator - browser-based interface for centrally managing Process Manager business processes. Compass Server - A profiling server that offers search, browse and profiling capabilities to help administrators gather and organize enterprise resources scattered across intranets so that users can find and retrieve information more efficiently. Media Server - An audio publishing, broadcasting, and receiving system that enables the creation and delivery of media-rich information, both inside and outside the enterprise. Media server includes four components: Media Server - play real-time audio feeds, provide on-demand access to pre-recorded audio clips, and synchronize audio with HTML documents, Java applets, and JavaScript applications. Media Proxy Server - a transparent

intermediary between Media Player and Media Servers which provides safe passage through the firewall for audio connections and operates as a reverse-proxy outside a firewall. Media Converter - compresses and converts different audio formats. Media Player - a plug-in needed to access audio files or a live feed from a Media Server.

Detailed Description Paragraph Table (4):

Product Name/ Category Product Details Business3 A software application that allows Business3 users NetMail to access their Business3 mail through a standard web browser without any Business3 software. Business3press A web publishing tool which may be published to any web server. Business3press offers the following capabilities: WYSIWYG editing Simple interfaces of creating forms and image maps Integrated browsing and editing simultaneously "Check Links" function to fix broken links Database interaction Permission setting Work archive MiniWeb - site management tool that provides graphical overview of website structure. It provides a mechanism to save or move multiple pages while maintaining appropriate links. Business3server A multi-threaded web and publishing server that provides the following capabilities: Serves HTML pages and other media files Runs CGI scripts and processes server-side includes Platform for dynamic web applications: Business3server Dynamic Pages (ADPs) Supports Business3server's C and Tcl scripting and APIs Supports database connectivity Allows users to edit content across the network with Business3press or other authoring tools Provides C API plug-in that can be used to serve and rotate web advertisements, as on Business3's site. Supports simultaneous connections through multi-threading and in-memory caching Supports site administration tasks including account management, document management (automatic version control and archiving), link management, and access control Web-based server and page administration Provides support for Art Technology Group's Dynamo server Business3server is used extensively on Business3's sites and a number of other Internet sites including the following: primehost.com, Business3.com, digitalcity.com, tile.net, am.net, worldpages.com: Client3 A software application that provides online chatting Instant capabilities, directory services for user profiles, Product1 and personalized news. Client3 A browser based upon Microsoft's Internet Explorer Browser which supports common internet services such as graphics, sound, meta-tags, plug-ins, security, FTP, HTTP. Client3 A software application installed on end-user's Client machines to obtain access to Business3's private network. Business3 Business communicates with a host in Virginia through a proprietary protocol. Client3 A server software that determines if a web page Caching object should be cached and when it should be Server check for a new version. This procedure, instituted in the Business3 proxy subsystem improves the performance of a website. Business3 Caching Server detects images and automatically compresses them for quick storage and access.

[First Hit](#)   [Fwd Refs](#)**End of Result Set**

Generate Collection

Print

L5: Entry 3 of 3

File: USPT

Nov 25, 1986

DOCUMENT-IDENTIFIER: US 4625081 A

TITLE: Automated telephone voice service system

Abstract Text (1):

An automated telephone voice service system includes a data store having a plurality of addressable voice storage message baskets defined therein and a control system coupled between the store and a large plurality of telephone lines of a telephone network. An incoming cable may address a particular message basket by entering a code through the telephone keyboard or by a predetermined association with a particular call in line. Upon identification of the message basket the caller is greeted by a client's own voice and invited to leave a voice message which will be recorded in the message basket or given other client information. Upon entry of a personal identification code a caller is granted access to user account functions which include retrieval of voice messages, forwarding of messages to other message baskets or telephone lines, and administrative functions such as the changing of greetings or account operating criteria. Editing commands may be utilized during the recording of voice messages.

Brief Summary Text (16):

Other telephone lines may be assigned as direct or general recall lines which afford a client access to account ownership functions afforded by the system. As with the incall lines each direct recall line is associated with a single predetermined message basket while a general recall lines requires entry of a message basket code identifying a desired message basket. Security is maintained by enabling account ownership activities only after a personal identification code has been entered which corresponds to an associated message basket. Added security may be implemented for a direct recall line by requiring entry of a second field of a personal identification code before account ownership activities are enabled. The second field is separated from the first field by a number sign key center and may be changed at any time by the account owner. Account entry thus requires a caller to have knowledge of the direct recall telephone phone number, the first field of the personal identification code associated therewith, and if used, the second field of the personal identification code.

Brief Summary Text (17):

Account ownership activities include retrieval of messages, forwarding of messages, and administrative functions such as the recording of a new greeting, the changing of answering criteria for a secretarial line or the changing of the second field of the personal identification code. Each message basket is divided into two parts, an inbasket which stores messages from outside callers and an outbasket which stores messages for forwarding to other inbaskets or telephone lines. Data storage space is conceived by storing only a single copy of an outgoing voice message in the client's outbasket, even if the message is to be sent to many different parties.

Brief Summary Text (21):

In the event a system user requires assistance, more detailed voice message prompts are initiated by keying \*0 and communication with a voice message operator can be commanded by keying \*20. In the event that a client calls the system from a dial

telephone, the service system detects a telephone company signal identifying a dial telephone line as the source of the call and automatically connects a service system operator to the line. The telephone service system in accordance with the invention thus provides a sophisticated user controlled system for the receipt and delivery of voice messages with an operator being required only for exceptional circumstances.

Detailed Description Text (7):

The prompts and client greeting section of data store 104 stores a plurality of individually addressable voice message prompts explaining how to operate the voice service system 100 and a client greeting for each inbasket. A voice message prompt is prerecorded for each anticipated state at which a caller might access the voice service system 100. These prompts provide an explanation as to how the user should proceed from the particular point of use and are accessed by the control system 102 and communicated to the user as appropriate. At any point, a knowledgeable user may override the prompt by inserting a command without taking the time to listen to a complete prompt message. The client greetings are provided as an answer mode for message storage accesses to each of the system inbaskets. Each client may record and change his own personal greeting at will. This enables the greeting to include current information such as telephone numbers at which the client can be reached for a given period of time, indications that the client is on vacation for a given period of time, indications as to when the client will return to his office and so forth. In the event that a client fails to have recorded a preestablished client greeting, a general system greeting is provided in its place. The system greeting invites the caller to leave a message but does not identify the specific owner of the inbasket which has been accessed by the call.

Detailed Description Text (111):

Two important principles should be noted from these examples, (1) The physical location of a process within the system 100 multiprocessor environment is not critical to the operation of the system and (a) all communications between system and user processes is by means of packet exchanges, even if processes happen to be co-resident in the same processor. Most of the Level 0 system functions and services are performed by a resident executive (REX) within the standard processor module of each system processor. A ROM copy exists in each processor to provide basic services to effectively manage the processor within which it is resident. The services include: interrupt handling, event management, timer management, memory management, process management, status monitoring, I/O service functions, list processing, inter-process communications, traps, wake-up and diagnostics.

Detailed Description Text (359):

The first three words of the 16-word data structure of a standard packet contain the routing information needed to identify the destination and the sending process. The two 20-bit process identification codes are packed into three words, each field having the following meaning:

Detailed Description Text (669):

(3) Event Management.

Detailed Description Text (675):

The event management routines enable a process to check for, and possibly wait on, a specified event. Events can be generated by both hardware and software and include interrupts, packet receipts, time-outs, I/O completion, process termination and signals from other processes.

Detailed Description Text (694):

The event management procedures deal with the detection of events and not the allocation or deallocation of event control blocks (ECBs). To the user of the ideal machine, event control blocks are an internal structure used totally by the system to maintain events for the user.

Detailed Description Text (695):

The procedures and functions relating to event management include:

Detailed Description Text (925):

IMMI and IMI functions are provided by a single wubprocess and use the suspend option when waiting for the completion of REX functions (I/O, Event Management, etc.)

Detailed Description Text (936):

(5) Process suspension and event management.

Detailed Description Text (1048):

D. Event Management

Detailed Description Text (1540):

In the event that the client has selected automatic answering service, the caller is greeted with a client selected greeting and invited to leave a message at the occurrence of a tone. The client may record his own greeting and change it at will, or alternatively, may use a system provided greeting which does not specifically identify the called client. Furthermore, the caller may specify the length of any message which can be recorded and the maximum number of messages which may be stored by the system. In any event, upon generation of the tone, the voice service system receives and records in the inbasket portion of the client message basket any message dictated by the caller. Up to the maximum message time specified by the client. During this recording process the system responds to message editing commands as if the caller were a system client. However, to avoid confusing nonclient callers, no editing prompts are provided and an unsophisticated caller may simply dictate a nonedited message with no knowledge of the system editing feature. Upon receipt of the message, the call is terminated and the line is released.

Detailed Description Text (1799):

Event management in the virtual machine consists of a set of interface routines to the higher level REX event management routines. Event management deals solely with the detection of events and not the allocation or deallocation of event control blocks (ECBs). To the user of the virtual machine, ECBs are an internal structure used totally by the system to maintain events for the user. The event management procedures and functions which the virtual machine provides include:

## CLAIMS:

17. An automated telephone voice service system comprising:

a store having defined therein a plurality of individually addressable message baskets, the store being coupled to store and retrieve representations of voice messages at each of the plurality of individually addressable message baskets therein; and

a control system providing a selective coupling between the store and each of a predetermined plurality of telephone lines of a telephone network, with the telephone lines including a direct incall line, the control system being responsive to different data signals received over a particular one of the telephone lines to associate the particular telephone line with a particular message basket, to store in the particular message basket a representation of a voice message received over the particular telephone line, and to forward a voice message representation stored in the particular message basket to at least one other of the individually addressable message baskets, and

the control system including means for detecting when the particular telephone line

is a direct recall line and responding to such detection by associating the particular telephone line with a predetermined particular message basket and precluding association of the particular telephone line with any other message basket, the control system being operable to enable an activity affecting the particular message basket only upon receipt over the particular telephone line of a predetermined personal identification code associated with the particular message basket when the particular telephone line is a direct recall line.

18. The automated telephone voice service system according to claim 17 above, wherein the predetermined personal identification code includes a first portion which cannot be changed in response to data signals received over the particular telephone line and a second portion which can be changed in response to data signals received over the particular telephone line.

20. The automated telephone voice service system according to claim 19 wherein the message basket indication is alternatively a message basket code or a personal identification code having a predetermined association with the particular message basket and wherein the control system responds to the message basket code by enabling a voice message recording with respect to the particular message basket or responds to the personal identification code by enabling account ownership activities with respect to the particular message basket and further responds to commands received as data signals over the particular telephone line by executing any activity commanded thereby.

33. The method of providing a telephone voice service system response to an incoming telephone call from a caller on a telephone line comprising the steps of:

communicating over the telephone line a prerecorded voice message prompting the caller to enter alternatively a message basket code or a personal identification code;

determining the type of code entered by the caller;

if a message basket code is entered, prompting the caller to communicate a voice message whose representation is forwarded to a message basket identified by the code and storing in a message basket portion of a store indicated by the message basket code a representation of any voice message communicated by the caller;

if a personal identification code is entered, enabling account ownership functions for an account associated with the personal identification code including retrieval of messages from a message basket associated with the account and forwarding of message representations from the associated message basket to another message basket identified by signals communicated over the telephone line in accordance with a predetermined code.

47. The automated telephone voice service system according to claim 43 above, wherein the data processing system includes means for processing a received data code which includes a first code identifying the owned account and a second, personal identification, code preceded by an ATTENTION command identifying the caller as the owner of the owned account.

49. An automated telephone voice service system comprising:

a store coupled to store and retrieve representations of voice messages at each of a plurality of individually addressable message baskets therein; and

a control system providing selective coupling between the store and each of a plurality of telephone lines of a telephone network with at least one of the lines being a general access line over which a plurality of different message baskets may be accessed for either message storing or account ownership functions, with a

message storing function being enabled in response to entry of a code identifying one of the plurality of message baskets and account ownership functions being enabled in response to entry of a code identifying one of the plurality of message baskets and a personal identification code identifying the owner of the one message basket.

50. The automated telephone voice service system according to claim 49 above, wherein the control system is operative to respond to a command series ATTENTION, CHANGE, CHANGE by enabling receipt of a different message basket identification code identifying a message basket different from a currently accessed message basket and granting access to the different message basket in response to the different code.

58. The method of telephone voice message communication comprising the steps of:

answering a telephone line;

receiving over the answered telephone line an identification code which identifies the caller as a subscriber having a subscriber message basket for storing data which includes representations of voice messages, the message basket having an inbasket portion and an outbasket portion;

receiving over the answered telephone line a first signal indicating at least one command including a talk command;

receiving over the answered telephone line and storing in the outbasket portion of the subscriber message basket in response to the talk command a representation of a voice message generated by the caller;

receiving over the answered telephone line a second signal including information identifying at least one designated recipient of the voice message; and

for each designated recipient:

calling the designated recipient by dialing a designated recipient telephone line corresponding to the designated recipient,

when the designated recipient telephone line is answered, communicating over the designated recipient telephone line a voice message delivery greeting including an explanation that a recorded voice message is being delivered,

retrieving from the outbasket portion of the subscriber message basket and communicating over the recipient telephone line the voice message, and

terminating the call to the designated recipient.

67. The method of telephone voice message communication according to claim 58 wherein the at least one designated recipient includes a subscriber having an identification code and an associated message basket having an inbasket portion and an outbasket portion and further comprising the step of storing in the inbasket portion of the message basket of the designated recipient subscriber information identifying the voice message and the location at which a representation of the voice message is stored.